FAMILYSEARCH API Feature Implementation Guide

Abstract

This Guide to assist in the design and flow of the screens and user experience. Sample screens for many steps have been provided.

Table of Contents

Authenticate

Desktop and Mobile App Users for FamilySearch Access

Desktop and mobile app users can authenticate with FamilySearch in order to access FamilySearch data. If the app is qualified to use a refresh token, then the authorization can be kept active for two weeks.

Assumptions

- You have an existing FamilySearch user account.

Programming Step

1. Obtain the user name and password. This can be done by inquiring of the user or using a name and password that your app has stored.
2. Display a permission request statement and obtain permission from the user.
3. Call the Access Token API resource. Specify the grant_type as "password" and provide the user name and password parameter values.

The user is authenticated when an access token is successfully returned to the app. This access token is then used as a parameter for all subsequent FamilySearch API resource requests.

User Interface Suggestions

1. Prompt the user for a FamilySearch user name, password, and display a permission statement.
   The user enters a user name and password and reviews the permission statement.
2. The user clicks LOGIN or CANCEL.
   The app requests and receives an access token, or cancels the login.
3. Proceed with the functionality of your app.

Authenticate Web App Users for FamilySearch Access

Users of Web apps can authenticate with FamilySearch in order to access FamilySearch Family Tree data. If the app is qualified to use a refresh token, then the authorization can be kept active for two weeks.

NOTE: It is a best practice for web applications not to store cached data in local memory storage, and to purge cached data at the end of each authorized session and whenever the back button is used.

This process of authentication is complete when an **access token** is obtained.

Assumptions

- You have an existing FamilySearch user account.

Programming Steps

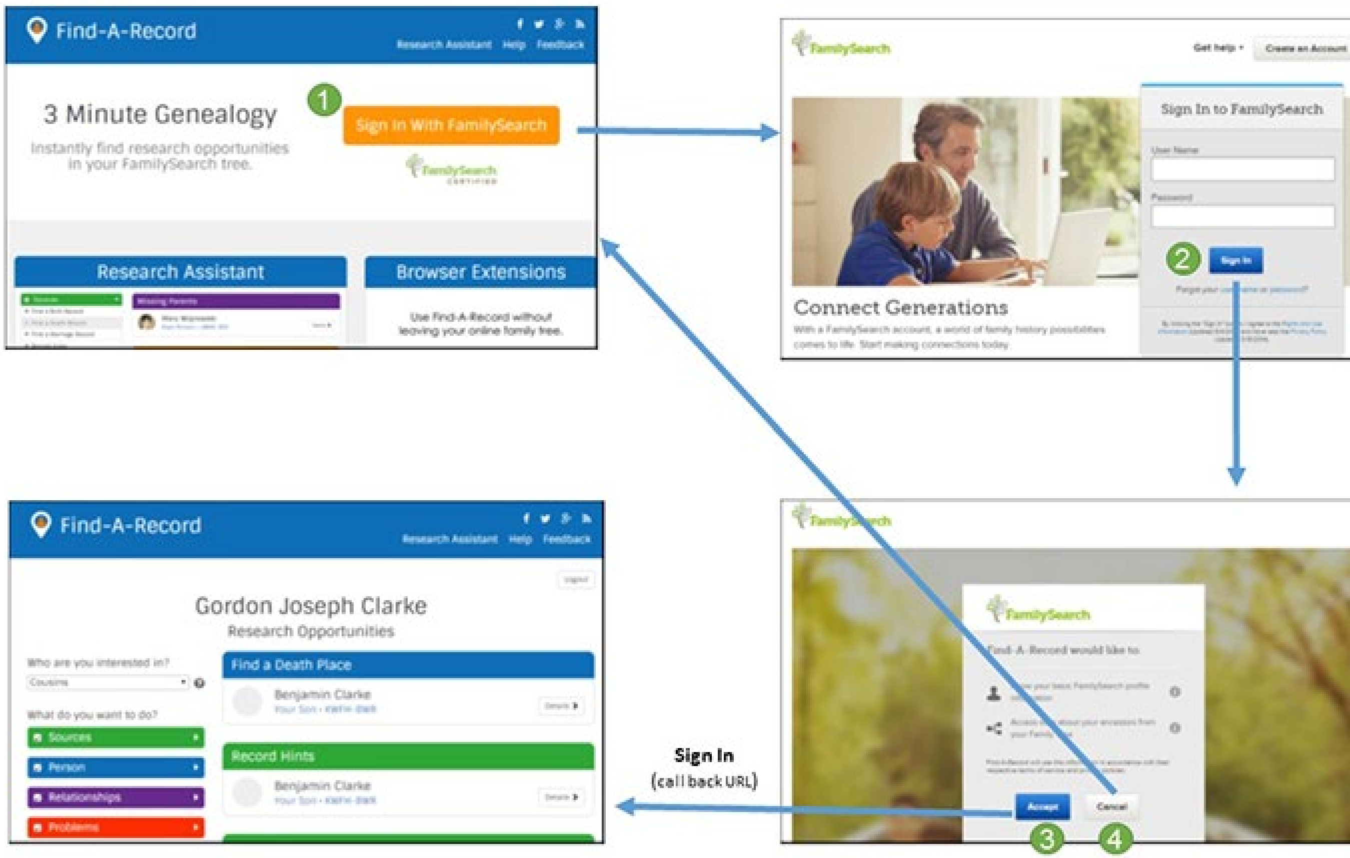
1. Call the Authorization API. resource. Provide a redirect URI parameter value such as http://localhost. The user is directed to the FamilySearch sign in page to provide FamilySearch access credentials (user name and password). An authorization code is sent back to the app.
2. Call the Access Token API resource. Specify the grant_type as "authorization_code" and place the authorization code you just retrieved as the code parameter value.

The user is authenticated when an access token is successfully returned to the app. This access token is then used as a parameter for all subsequent FamilySearch API resource requests.

User Interface Suggestions

1. Prompt the user to sign in with FamilySearch.
   The user clicks the Sign In with FamilySearch button and the app calls the authorization API which goes to the FamilySearch Sign In page.
2. The user provides user name and password and clicks the Sign In button. FamilySearch presents the user with the Permission Page to accept or cancel the transfer of FamilySearch profile and ancestors data to the app.
3. If the user clicks the Accept button, the user is authorized and the app goes to the redirect URI with an authorization code.
   The app calls the Access Token API to obtain an access token.
   Authentication is complete.
4. If the user clicks the Cancel button, the app goes to the redirect URI with the return string
   "error=access_denied&error_description=User+declined+consent". An authorization code is not included and authentication cannot be established.
   The application should present an appropriate response to the user.

The following screen shots are for a company called FindARecord.com.



Compatible Review Process

FamilySearch will review the software you have created to determine if it is compatible with FamilySearch technology and its core mission. This review is required in order to obtain a Beta or Production API key. Only a legal, registered business will be eligible for verification and listing in the App gallery. Sole Proprietorships will not be eligible

You must complete the preliminary steps outlined in the Getting Started Overview in order to proceed with this compatible review process.

Solutions should be "Compatible" for ***commericial distribution***.

There are two main types of compatibility you should be aware of:

- **Read**. There are many options available to become Verified Compatible for reading FamilySearch data. However, your solution must be verified with at least one read option before Verified Compatible to write.
- **Write**. Your app can Verified Compatible to add people, update people, write sources, write memories, or write discussions.

In addition some apps will be certified for special functions such as LDS ordinance work or bulk record handling.

The Business Approval Process

The business qualification process consists of the following steps:

1. Apply for read compatibility using the “Apply for Compatiblity” button within the “My App” section of the developers “Logged in” experience. You will receive an email with instructions on how to proceed.

2. Complete, sign, and return the Compatible Product Affiliate Agreement, Security Accessment, and the Production App Key Request and Use Agreement.

3. Register your solution with FamilySearch to monitor your solution compatiblity verification progress.

4. Develop your solution to read data. During development you can view the integration data through the FamilySearch web client at http://integration.familySearch.org. Sign in with the integration user ID and password you received when you registered as a FamilySearch partner.

5. Your solution will be evaluated as described below under Solution Compatiblity Evaluation. After your solution is accepted, you will receive a production app key and you can add yoursolution to the App Gallery.

6. Your solution will be listed as "Compatible" in the App Gallery.

   Note: After your solution is compatible to read, you may then proceed to apply for Write compatible, if you solution will be writing data to FamilySearch.

7. To apply for Write compatiblity, when your app is near completion send an email to devsupport@FamilySearch.org. You will receive instructions on how to apply for write compatiblity, and how to modify your solution for new features in the FamilySearch App Gallery. If you and FamilySearch mutually agree on the need for beta testing before going public your app will be given access to our beta test data. During beta testing, your app users will access the beta database by signing in to http://beta.FamilySearch.org. You will need to modify your solution to authenticate for access to the beta database instead of the integration database. See Authentication.

Required Agreements! Your solution will not be Verifed Compatible until you have received from FamilySearch the approved and signed copies of your Compatible Products Agreement, and your Use Agreement.

Reverification Requirement! When a solution is changed to alter the way it adds, modifies, or deletes Family Tree information, those changes must be Verified Compatible before releasing the changes to the public.

Warning Concerning Undocumented APIs! An undocumented API is an API that is not documented on the FamilySearch developer website. There is no support for undocumented APIs and there is no guarantee of the behavior or longevity of undocumented APIs. The use of undocumented APIs may create a poor user experience and affect your product stability. A solution that uses undocumented APIs will not be listed in the App Gallery, will not enjoy the benefits of being verified compatible, and the app key issued for that app will be disabled. The continued use of undocumented APIs will jeopardize any business relationship with FamilySearch.

Solution Alignment Guidelines

In addition to the technical compatibility requirements needed for accessing the FamilySearch API as set forth in the compatibility check list for each set of features, the solution will be reviewed for its alignment with the core mission of FamilySearch and the capabilities of FamilySearch APIs.

Solutions ***most*** likely to be approved include those that:

1. Facilitation and contribution to FamilySearch Family Tree in the form of persons, vitals, records, and memories while following Source-Centric Open Edit (SCOE) principles.
2. Assistance with the research process.
3. Analysis and graphical representations not available from FamilySearch.
4. Applications that engage the user in the research or ancestor discovery process.

Solutions ***least*** likely to be approved include those that provide:

1. Appear to only read FamilySearch content without creating meaningful value to the FamilySearch website or its contributors (so, be sure to clearly explain the benefits of your solution!).
2. Put an unusual load on FamilySearch's ability to service FamilySearch and other third-party solutions.
3. Duplicate capabilities already available from FamilySearch.

Please Note:As of June 27, 2017, FamilySearch is suspending the certification and approval of new applications that provide functionality for scanning, capturing or reporting (or any combination of the three) of ordinance information, including those that are or may be available to take to the temple. As we evaluate the impact of these types of applications on the FamilySearch systems and the end-user experience; this suspension will remain in effect until further notice. Solutions that request limited read-access privileges may be considered based upon need and functionality.

Solution Compatibility Evaluation

Each app submitted for certification will be evaluated on the following criteria.

Product Review for Read Solutions

Read compatiblity for solutions includes the following product review:

- Reviewing screen shots of the product, a recorded demonstration of the product, or a working URL beta site.
- A product review board examination of general usability of the product for its intended audience. Required features for a given solution or utility will be checked to see if these are in compliance. This is usually done by a demo and a live interview.

Important! A review is performed separately to certify each type of read operation your app performs. See the Read Compatiblity Checklist.

Product Review for Tree Write Solutions

Tree Write certification for apps includes an extended product review, a technical review, and a security review. Sometimes the business, product, and technical review can be done in one meeting. The security review is always a separate activity scheduled and performed by the security team.

Extended Product Review

As with a read only app, the product review board examines the general usability of the product for its intended audience. The required features for a given application or utility are checked to see if these are in compliance. The app is also evaluated for adherence to Source Centric Open Edit (SCOE) standards. This usually requires a demo followed by a live interview.

Technical Review

Technical reviews are used to determine what programmatic calls are being made to get various responses presented through the user interface. Various use cases are often tested to make sure the logic is in place to accommodate most possibilities. Sometimes the product review and the technical review are done in the same meeting. Applications for write certification usually take multiple meetings to see everything and go through all of the use cases that can be encountered when interfacing with a large collaborative family tree.

Security Review

A security check of web, desktop, and mobile applications is done by FamilySearch software security personnel. These are done by request, and the partner does not attend or participate in any way except to provide the app URL if it is a web app, the installable software if it is a desktop app, or a mobile device with the software already installed if it is a mobile app. The reviewed items are covered in the authentication requirements for web, desktop, and mobile.

Compatibility Compliance

Solution providers and their solutions(s) that have write access to production data through an issued App Key will be checked from time to time to see if they are in compliance 1.)Compatibility requirements that were effective at the time of their compatiblity verification or 2.) new requirements that have been effective according to the postings in the Change Log of the FamilySearch.org/developers website. When the solution provider adds new functionality or changes the functionality of features that write data, they must re-verify this feature.

FamilySearch Compatible Logo Use Requirements

The “FamilySearch Compatible” logo must be displayed within the web, desktop, or mobile solution. This logo can be used with sign-on screen for desktop and mobile solutions using “password flow” method for authentication. It can also be used for promotional purposes on websites and other marketing material to promote the status of the solution. In all cases, the use of the “FamilySearch Compatible” designation should include a rollover message or link to the FamilySearch Compatible disclosure statement. Printed material displaying using the “FamilySearch Compatible” designation or logo must have an asterisk to reference the full disclosure at the bottom of the page.

FamilySearch Compatible Disclosure

“FamilySearch Compatible” designation and logo is used to identify software solutions that FamilySearch International (“FamilySearch”) believes to be generally compatible with FamilySearch.org or one or more of FamilySearch’s APIs (application programming interfaces). FamilySearch, however, takes no responsibility and is in no way liable for any such solution. Accordingly, FamilySearch in no way warrants that any solution associated with this designation or logo will function as intended, is free from harmful or undesirable aspects, or is free from errors.

Non-Compliance Situations

If the app or app provider is found to be out of compliances, measures will be taken to motivate the app providers to bring the app back in to compliance. This includes the following situations:

1. Using a feature that has never been Verified Compatible.
2. Issuing a new release of a previously compatible feature that is no longer in compliance.
3. New Verified Compatible requirements have not be implemented within an expected timeframe.
4. 4. Solution provider fails to meet contractual requirements regardomg:
    - Availability and effectiveness of support
    - Billings and collection policies
    - FamilySearch Logo and Trademark Usage Guidelines”
    - FamilySearch API License Agreement” **(See Note 1 below)**
    - Data Quality Agreement (when this becomes available)
5. Availability of working solution
    - Desktop or mobile solution
        - Download or installation process not working
        - Unrepaired broken desktop/mobile solution
    - Web solution
        - Subscription process not working
        - Frequent downtime or unacceptable performance

Determining Non-Compliance

In addition to periodical compliance auditing of existing certification requirements, FamilySearch will check compliance on those applications that are affected by new compatibility requirements according to the effective date posted in the Change Log. FamilySearch will also respond to any reports of non-compliance by employees, partners, or customers. If after reviewing the desktop, mobile, or web solution the non-compliance and timing is validate, FamilySearch will make initial contact by phone or email to comfirm that partner understands the non-compliance will be resolved quickly (5-7 days). If non-compliance is not resolved within this timeframe, the Compliance Motivation Steps may begin as follows:

Compliance Motivation Steps

Step	Timing	Action
0	Inital Contact	Informal notification and request by phone or email to quickly (5-7 days) remedy **(See Note 2 below)**
1	10 days later	Non-compliance notification letter sent with “Compliance Motivation Steps”
2	30 days later	Notice put in the Product Detail page and sent to the Sp;itopm Provider “Solution is no longer compa FamilySearch”
3	30 days later	App is pulled from the App gallery and notice sent to the App Provider
4	30 days later	App Key is turned off until sufficient evidence is provided concerning how the non-compliance has be
5	TBD	App Key is turned on after correction is validated

Note 1 - Termination Section 6 of FamilySearch API License Agreement

FamilySearch reserves the right to terminate this Agreement or suspend or discontinue Your access to the API, or any portion or feature thereof, for any or no reason and at a time with or without notice to You and without liability to You. Upon the early termination of this Agreement, Your App Key shall be revoked, and all licenses granted hereunder shall terminate.

Note 2 - Prompt Resolution

Typically, non-compliance issues can be resolved quickly without triggering the “Compliance Motivation Steps”. Please respond promptly to any communication you receive from FamilySearch.

Data Management

The volume, accuracy, security, and interoperability of the data in the FamilySearch Family Tree is extremely important to customers of FamilySearch and FamilySearch partners. The responsibility for data management is shared and should follow established best practices.

Shared Responsibility

The main attraction to FamilySearch is the value of its Family Tree and historical records to customers. Inaccurate, inappropriate, and damaging data can degrade the usefulness of the FamilySearch website and solutionis that consume FamilySearch data. Data management must be a collaborative effort between FamilySearch and FamilySearch solution providers, assisting users in how they complete, submit and report abuses of information. Everyone benefits from working together to manage the data in FamilySearch.

Data Quality Practices

All solutions should employ the following minimum data quality practices.

- Solutions that capture data through form entries should guide the user on the intent, length, and acceptable format of entered information.
- Solution should identify inappropriate values and give the user the opportunity to re-enter appropriate information.
- Solution should follow industry best practices for security related to both software and data. For example, consider owasp.org for web app security to prevent attacks, code injections, and other vulnerabilities.
- Solutions should provide methods for users to report data inaccuracies and information abuse to FamilySearch.

Data Security Practices

All solutions must employ the following data security practices.

- Solutions can only show living and ordinance information to the FamilySearch authenticated user who requested the information.
- Deceased non-ordinance information can be shared with any FamilySearch authenticated user.
- Local browser cached data from FamilySearch must be cleared at the end of the browser session. Likewise, requesting browser history or using the browser back button cannot reveal FamilySearch data that was previously accessed.
- Long Running Task Completion is allowed with the following limits.
    - Periodic checking for person updates is not a single long running request.
    - The FamilySearch session must be expired after the user is gone except when the user starts a single long running request. A single long running request does the following:
    - `- Imports a reasonable number of generations (initial pulls).`
    - `- Checks for needed temple ordinance work.`
    - `- Updates person information at the user's request (not scheduled).`
    - `- Does not make additional user requests after the user is gone.`

      ```
      - Allows all requests to finish that are initiated by users while they a
      re online.
      ```
- Solutions can save genealogical information of deceased individuals retrieved by FamilySearch authenticated users.
- Solutionscan save but cannot publicly share genealogical information of living ancestors retrieved by FamilySearch authenticated users.

  <="" li="">

    - Living ancestors can only be seen, modified, and managed by the authenticated user in their own private space.
- Solutions can save FamilySearch person ID numbers of persons needing ordinance work, but not the ordinance data. Persons needing ordinance work cannot be shared with users who have not already reserved the person. In order for a different user to view a saved person ID needing ordinance work, the user must have the permission to view LDS information.
- Solutions can cache boolean metadata indicating which ordinances are needed. The app **cannot** display this metadata to the user until the "View LDS Information" permission is verified for that user session (once per new session).
- Solutions cannot store any LDS Ordinance dates or places as a "Presence Event" to show that the individual was present in a specific location on a specific date.

Church Related Services

Overview

Compatible Review is a process that a vendor's third-party application meets the performance and quality assurance requirements stipulated by the reviewing entity. FamilySearch reviews compatibility of minimum capabilities for different sets of features to make it easier for consumers to understand what features a vendor offers. Soltuions that provide ordinance services should comply with all compatible capabilities as presented in the Ordinance Compatibility Summary section.

Please Note: As of June 27, 2017, FamilySearch is suspending the compatibility verification and approval of new applications that provide functionality for scanning, capturing or reporting (or any combination of the three) ordinance information, including those that are or may be available to take to the temple. As we evaluate the impact of these types of applications on the FamilySearch systems and the end-user experience; this suspension will remain in effect until further notice. Solutions that request limited read-access privileges may be considered based upon need and functionality. If you have any questions, please email devsupport@familysearch.org.

Identify Appropriate Permissions

The application must make the proper call to the Identity module of the API to determine the privileges of the authenticated user.

Pricing of Temple Related Features

Temple attendance and the performing of temple ordinance is a sacred religious activity for the members of The Church of Jesus Christ of Latter-day Saints. Therefore, FamilySearch will review the pricing model of products that contain features specific to The Church of Jesus Christ of Latter-day Saints members. There should not be an additional cost for temple related features. If a software provider has an application that only provides these features, that application should be free. If the software provider has a business model for “freemium” and “premium” pricing, the temple related features should be provided in all versions offered. It is permissible for the solution to ask for donations including suggested amounts.

Adjustable User Interface

The user interface available to authenticated users needs to be adjustable, depending upon the privileges that are available to the general public verses an church member. The general public should not see any icons, menus, labels, or messages that are intended for The Church of Jesus Christ of Latter-day Saints members. Solutions that are verified compatible for Ordinance Read, Analysis, Reserve, or Family Name Cards should have these features turned off for public authenticated users.

Ordinance Compatibility Summary

All solutions accessing ordinances must meet the following business, marketing, and privacy requirements:

- All public communications about ordinance features should be reviewed and approved by FamilySearch before being distributed by the certified vendor.
- FamilySearch reserves the right to request additional reviews periodically or when the vendor's application goes through a version change.
- Vendors should submit product names and Internet domain names for approval by FamilySearch before publicly deploying a product with ordinance capabilities.

A solution can be verified compatible in the following ordinance features: Identify Opportunities, Reserve, and Print Family Names.

Comparison Chart

Required: Yes **Not Allowed:** No **Optional:** Opt

	Identify Opportunities	Reserve	Pri
Prerequisites			
Previous Certified	Tree Read	Tree Write	Tre
Ordinance Addendum	Yes	Yes	Yes
Church Account	Yes	Yes	Yes

Required Features			
Person Ordinances Read	Yes	Yes	Yes
Identify Persons Needing Ordinances	Yes	Yes	No
Checking for Duplicates	Opt	Yes	Yes
Prepare Persons for Submission	No	Yes	No
Planning a Visit	No	Yes	No
Printing Family Ordinance Requests (FORs)	No	Opt	Yes
Use of Icons and Statuses	Yes	Yes	Opt
Use of Submission Policy	No	Yes	No

Identifying Opportunities

Solutions that show ordinance information and status along with other person information need to properly use the Ordinance Person Read endpoint. To identify opportunities, applications must crawl the tree to gather information and statuses about ancestors.

These solutions not only crawl the tree to gather ordinance information but also perform the analysis to find couples, parents, and children that can be submiited for ordinances with very little or no additional information. The results of this crawling and analysis can be shown in an "Opportunity List," which indicates which individuals need ordinances and the actions required to add the names to the user's Temple Reservation List.

Reserving Ordinances

To receive this compliance verification, solutions add persons to the Temple Reservation List, display the Reservation List, and select individuals and ordinances to be reserved. The Temple Reservation List should show at least the name, ordinances reserved, who the person needs to be sealed to (if applicable), and the date reserved.

Ordinance status icons should always be used on the Temple Reservation List to show status of all ordinances reserved for individuals. The ordinance icon legend should be easily displayed from this page. The minimum actions that must be available are to Reserve, Unreserve, and Share.

Requesting Ordinances

This feature makes it possible to select individuals and some or all ordinances that have been reserved in order to place these requests on a Family Ordinance Request page or Temple Cards. The solution receives the Family Ordinance Request (FOR) or Temple Cards in PDF format from FamilySearch.org and launches Adobe Acrobe Reader or other compatible reader. The solution should also allow lost or destroyed requests to be reprinted.

Identify Persons Needing Ordinances

Helping users determine whether a person needs temple ordinances is a task that applications can distinguish themselves from the FamilySearch Family Tree web interface and other products.

Issues

Church members who have many generations of data in Family Tree may find it difficult to find ordinances that can be done. They do not necessarily want to navigate through their family tree to find them. Programmatic crawling and analyzing ordinance statuses can help members decide what persons can more easily qualify for ordinance submissions.

Minimum Requirements

- Make sure a person's record is in Family Tree before it is reserved. This ensures that official ordinance records can be used to determine whether ordinances are needed.
- Use the Person Ordinances Read service to identify the ordinance statuses for a person or family.
- Display ordinance statuses using the approved definitions and icons.

Prepare Persons for Submission

When users reserve ordinances, they accept the responsibility to see that the ordinances get done. After reserving ordinances, users assign them to an inventory. The inventory indicates who will provide the proxies for the ordinances.

Issue

Users need to reserve ordinances that they want to do. Users may need help from others to get the ordinances done.

Minimum Requirements

- Reserve only ordinances for which the status is listed as reservable.

- Ensure that all prerequisite ordinances are either done, not needed, or reserved at the same time.
- Allow users to reserve ordinances for themselves or share them with the community.
- Allow an ordinance that was previously shared to be reserved, if the work is not yet done.
- Allow an ordinance that was previously reserved to be shared.

Check for Duplicates

Preventing the performance of duplicate temple ordinances is one of Family Tree's main purposes.

Issues

To prevent duplicate ordinances, a person's record must be merged with the records that contain the official ordinance assertions. This may create the following challenges:

- End users may become discouraged or bored if they have to review too many possible duplicates.
- Users may be confused if they merge some duplicate records while submitting ordinances and then later find that more possible duplicates exist for the same person.

Minimum Capabilities

- Show a merge screen to the user. This screen can appear before or after the screens where users identify persons needing ordinances and where users prepare and assign ordinances. To do this, an application must have the Merge certification.
- Reread ordinance information after records are combined to obtain the new ordinance status.

Printing Temple Ordinance Cards

Temple Ordinance Cards can be printed for the persons for whom they will be performing ordinances.

Issue

Church members generally print Temple Cards when they create a temple trip. However, the printing can occur any time after the trip has been created.

Minimum Capabilities

For printing and reprinting Temple Cards, the following requirements apply:

- After ordinances are reserved, users should be able to find, select, and print reservations on temple cards.
- The solution should provide a list of people whose ordinances were previously printed on cards and give users the ability to reprint a temple card for any of those people who still qualify for ordinances.
- If a card is reprinted after some of the work has been performed, the solution should only reprint the card with the ordinances that have not been completed.

- After printing a card, the solution should ask if the cards printed correctly.
  - If yes, the process is completed.
  - If no, the user should be prompted to print again.

Displaying Ordinance Status

Ordinance Abbreviations

The abbreviations B, C, I, E, SP, and SS represent the names of each of the temple ordinances in English. The detailed descriptions of the abbreviations can be shown to authenticated members. The same abbreviation letter can be used in all languages.

Letter	Ordinance	Prerequisites
B	Baptism	None
C	Confirmation	Baptism
I	Initiatory	B, C
E	Endowment	B, C, I
SP	Sealing to Parents	B, C, I, E
SS	Sealing to Spouse	B, C, I, E

Ordinance Status Colors

When the solution displays a deceased person’s ordinances, it should show display the ordinance status and if applicable the completion date. The ordinance status is a colored box with the appropriate color and ordinance abbreviation letter inside. Additional status text may be used along with the color.

Color	Hex Code	Status Codes

Gray	`#888a8c`	`http://familysearch.org/v1/BornInCovenant`
		`http://familysearch.org/v1/Completed`
		`http://familysearch.org/v1/NotNeeded`

Green	`#75bb49`	`http://familysearch.org/v1/Ready` `http://familysearch.org/v1/ReservedSharedReady`

Blue	`#47acd7`	`http://familysearch.org/v1/Reserved`
		`http://familysearch.org/v1/ReservedPrinted`
		`http://familysearch.org/v1/ReservedWaiting`
		`http://familysearch.org/v1/ReservedShared`
		`http://familysearch.org/v1/ReservedSharedPrinted`

Orange	`#f5ae2f`	`http://familysearch.org/v1/NeedMoreInformation`
		`http://familysearch.org/v1/NeedPermission`
		`http://familysearch.org/v1/NotAvailable`
		`http://familysearch.org/v1/NotReady`

Considerations for Mobile Solutions

Mobile solutions are easier and have shorter tasks to accomplish. The follow simplifications are acceptable with mobile solutions

- The status letter displayed without square colored background.
- The letter has two states for available and unavailable using a combination of light, standard, or bold styles.
- The letter “S” is used for both “Sealing to Parents” and "Sealing to Spouse” as long as the actual status of each one of these is easily accessible.

Use of Submission Policy

110 Year Rule

To do ordinances for a deceased person born in the last 110 years, members must either be one of the closest living relatives or must obtain permission from the closest living relative. Church members should not submit the names of persons who are not related to them. The ordinance submission application must remind the church member of their responsibilities and give them the opportunity to accept these reponsibilities.

Acceptance Process

1. Display the Church Policy screen.
2. If an ancestor was born in the last 110 years, request the user to indicate that he or she is authorized to perform the ordinances:
    - If you are the spouse, parent, child, brother, or sister of the deceased person, click **I am the closest living relative**. You can request the ordinances.
    - If you are not the spouse, parent, child, brother, or sister of the deceased person but you have permission from the closest living relative, click **I have permission from the closest living relative to do the ordinances**. You can then request the ordinances.
    - If you are not a close living relative and if you do not have permission from the closest living relative, click **None of the above. (This person's ordinances will not be reserved.)**
3. Also display "For more information, please read (including proper hyperlinks):
    - Chapter seven of *A Member's Guide to Temple and Family History Work*.
    - Terms of Use.
    - The First Presdency message on *Names Submitted for Temple Ordinances*.
4. At the bottom of the screen, provide a check box for users to indicate that they have read and will comply with the Church Temple Ordinance Policy. Clicking the check box means the users understand Church policy and are reserving only ordinances that comply with the policy.
5. Provide an **Add** button to put the person on the user's temple list. If a user clicks it, reserve the ordinances, and display the temple list.
6. Provide a **Cancel** button that does not put the names on the user's temple list but instead returns the user to the prior screen.

For links to translations of the required links, please contact developer services.

Submission Policy

Since the Submission Policy may change, it will be a resource obtainable as an API resource with an active session and autheticated member. The instructions for accessing this resource will be made available to developers that have signed the "Ordinance Addendum."

Read Implementation Guide

Read Change History

The change history for a person or relationship can be read. Each entry in the change history contains details about the change.

Assumptions

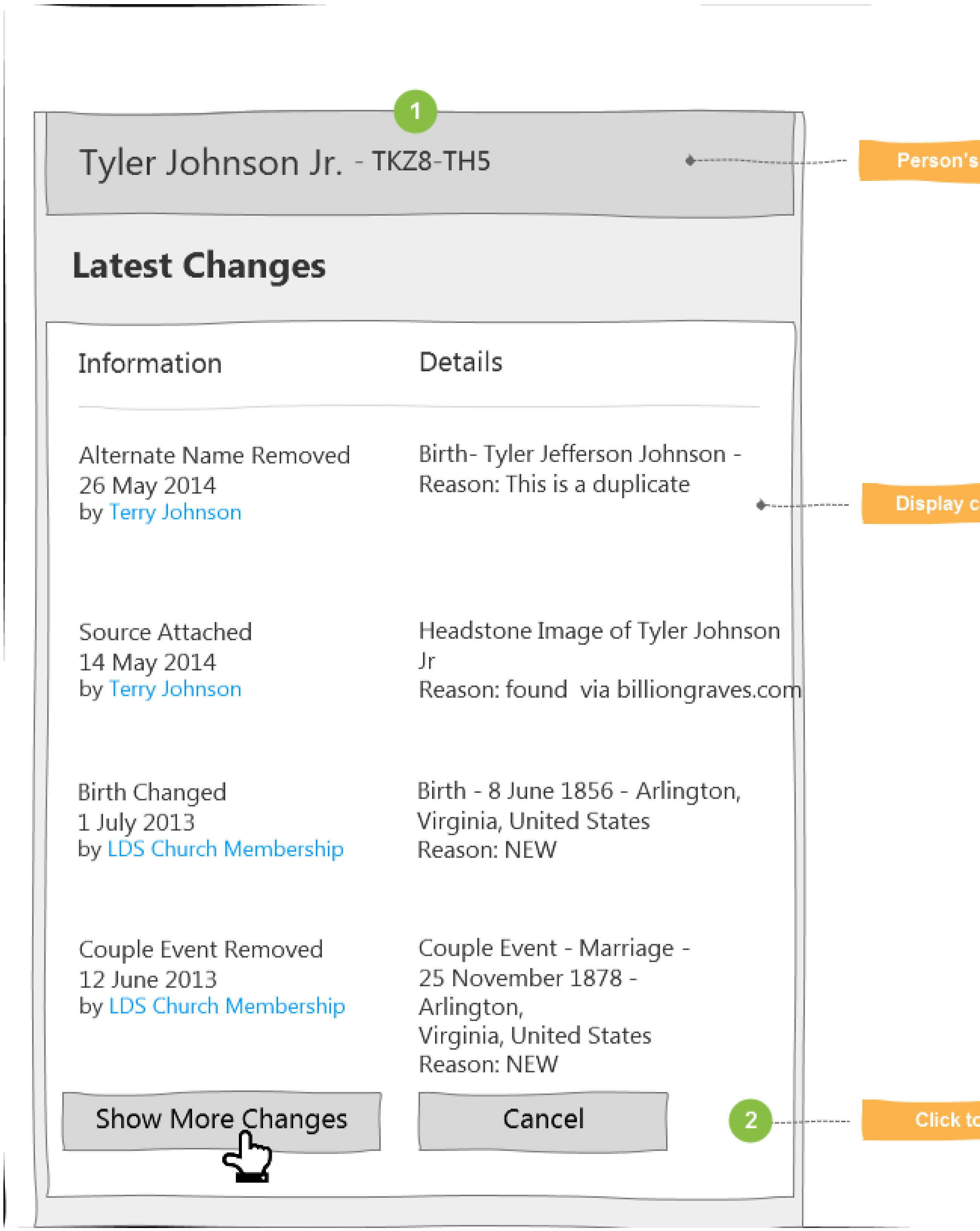
- The system must have some logged history regarding a single person or relationship.
- You must have the ID of the person whose history you want to read.

Programming Steps

1. Read Person Change History
2. Display the Change History Log.

User Interface Suggestions

1. Display the following information for each change history item of the selected person.
    - Title
    - Description
    - Date
    - The user who made the change with a link to the user profile
    - Reason for the change
2. Limit the number of Change History items displayed to an initial maximum value. Allow the user to view more until all items have been listed.



Additional options not required for certification may include the following.

- For each Change History item, offer to restore the state of the person or relationship to an earlier state. See Change History and Person Restore in Family Tree.
- Sort changes by date.
- Provide a link to FamilySearch using "/#view=personChangeLog&person=[PID]" from a Person Summary or Person Detail Page where the [PID] is the FamilySearch Person ID (XXXX-XXX). This URI is subject to change. FamilySearch may be providing a more persistent entry point to the FamilySearch page that will utilize ARK links.

Read Discussions

Discussions and comments that have been created for a person in the FamilySearch Family Tree can be read. The discussion details and comments included in the discussion can be listed. The details include the description, the contributor, and the contribution date.

Assumptions

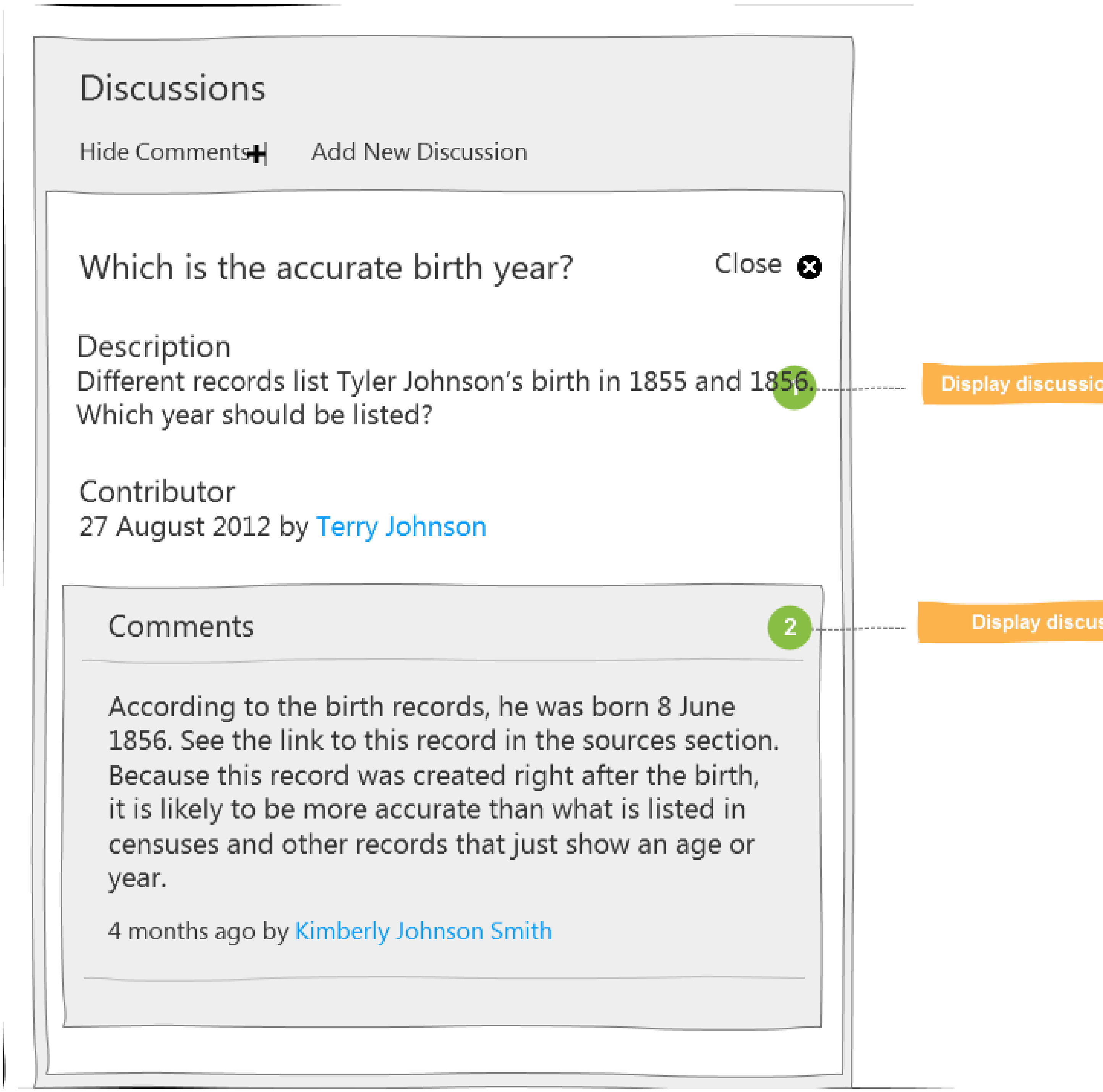
- You have the person ID of the person whose discussions you want to read.
- A discussion with content has been added to the target person.

Programming Steps

1. Read a discussion.
2. Read the discussion comments, and display the content.
3. Repeat for remaining discussions.

User Interface Suggestions

1. Display the descriptive information of a discussion.
2. Show or hide the content of the discussion.



Record Hinting

After associating a person in your application with a FamilySearch Family Tree Person ID (PID), this PID can be passed to the FamilySearch Hinting API to receive a list of historical records matching the vital and relationship data of that person. FamilySearch provides a page where the user can review historical records and optionally attach them to the FamilySearch person. You can then use the FamilySearch API to retrieve the newly attached sources and any newly added persons.

Programming Steps

1. Search for hints using the Person Match FamilySearch API resource. You must specify the collections query parameter in order to obtain hints from the hinting system.
2. Redirect to the ***Possible Matches*** detail page on FamilySearch.
   https://familysearch.org/ark:/61903/4:1:{PID}?context=recordHints
   or

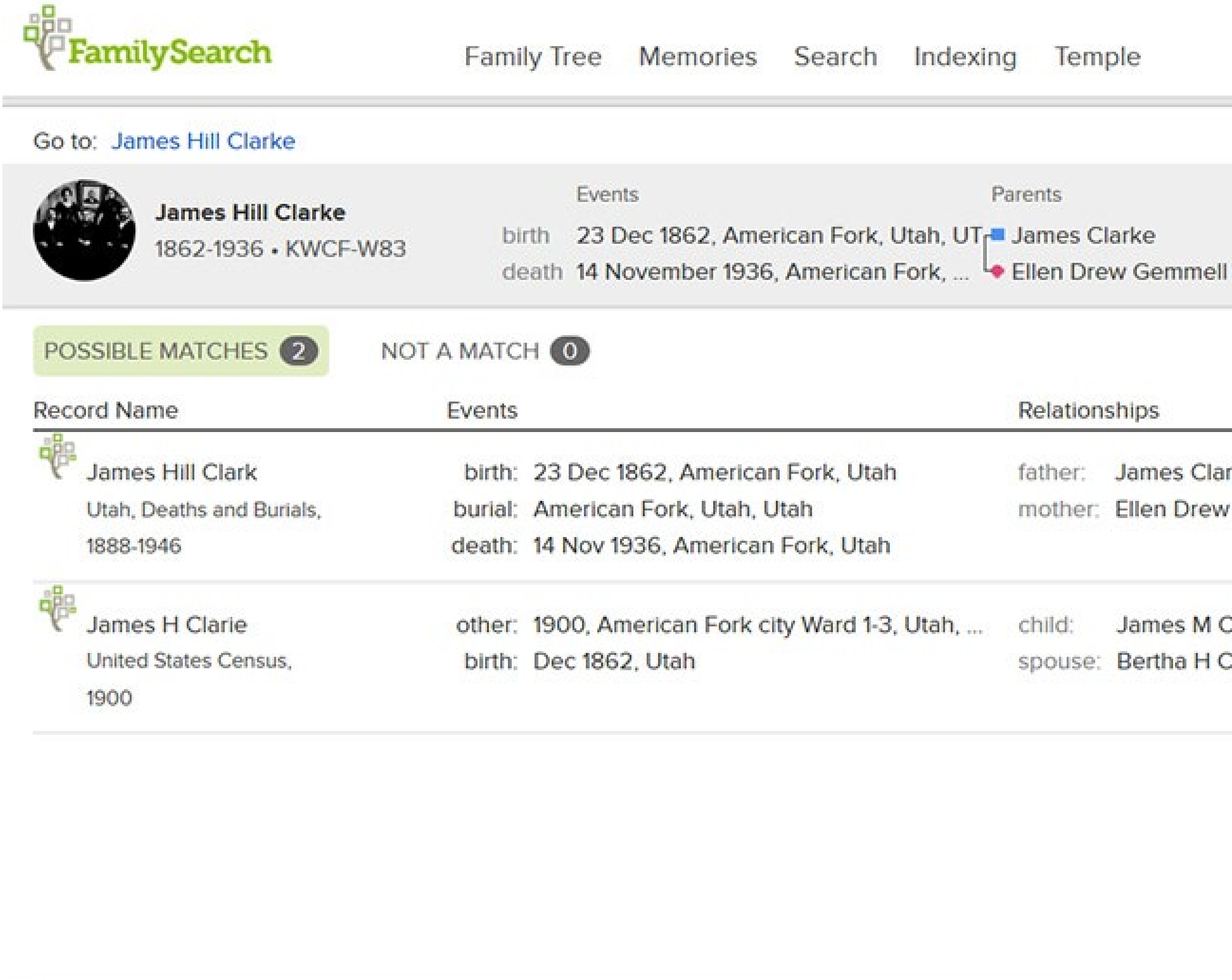
3. Redirect to the ***Attach Historical Records to Family Tree*** page on FamilySearch.
   https://familysearch.org/ark:/61903/4:1:{PID}?context=sourceLinker&hintId={hintURI}

User Interface Suggestions

OPTION A: Redirect the User to the Possible Matches *Detail Page on FamilySearch*

1. Display an option to initiate a hints request or automatically perform hints requests at predetermined points in the application.
2. Notify the user that record hints have been found as a result of a hints request.
3. The user requests to see the hints and is redirected to the ***Possible Matches*** detail page. A message could be displayed indicating that a FamilySearch page will be

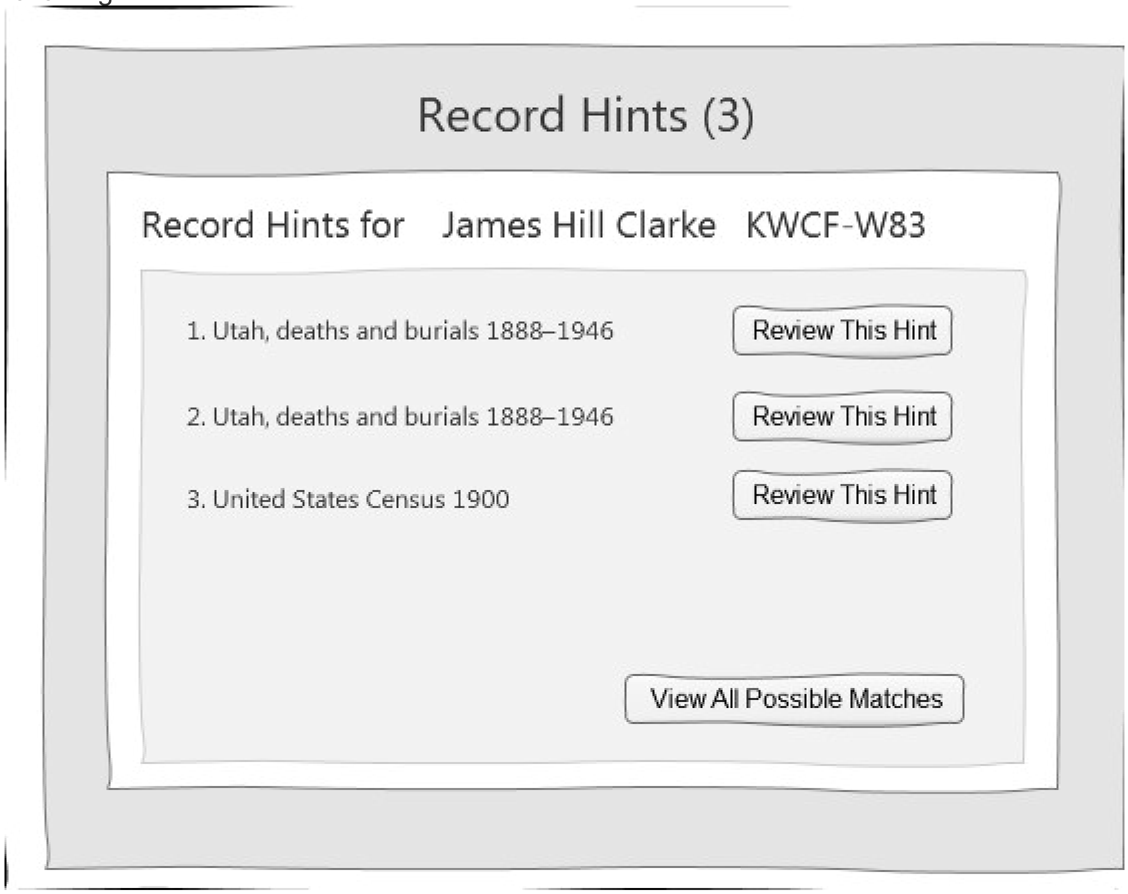
opened.



OPTION B: The Application Provides Its Own Record Hints Page

1. Display an option to initiate a hints request or automatically perform hints requests at predetermined points in the application.
2. Notify the user that record hints have been found as a result of a hints request.
3. The user requests to see the hints.

4. Create and present to the user a screen substantially similar to the following.



The 4 or 5 star rating for each hint could also be shown.

5. For "View all Possible Matches", direct the user to the ***Possible Matches*** detail page on FamilySearch. A message could be displayed indicating that a FamilySearch page will be opened.
For "Review this Hint", direct the user to the ***Attach Historical Records to Family Tree*** page on

FamilySearch.

Read Memory Comments

Comments on a memory can be read and displayed to the user.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- The user has selected a memory to view the comments.

Programming Steps

1. Prompt the user to view the comments of the selected memory.
2. Read comments on the memory. (Memory comments are also available by a link in the reply to reading a memory.)

User Interface Suggestions

1. Display a list of memories and allow the user to select a memory.
2. The user selects an option to display the memory comments (or cancel).
3. Display the comments.

Related Topics

- Read a list of memories
- Read a selected memory

Read Memories Attached to a Person in FamilySearch Family Tree

Users can view a list of memories that are attached to a person in FamilySearch Family Tree. The following details of a memory can be displayed:

- Title
- Description
- Contributor
- Time stamp
- Other metadata
- Comments

Assumptions

- At least one memory has been attached to the FamilySearch Family Tree person whose memories you want to view.
- You know the person ID (PID) of the person whose memories you want to view.

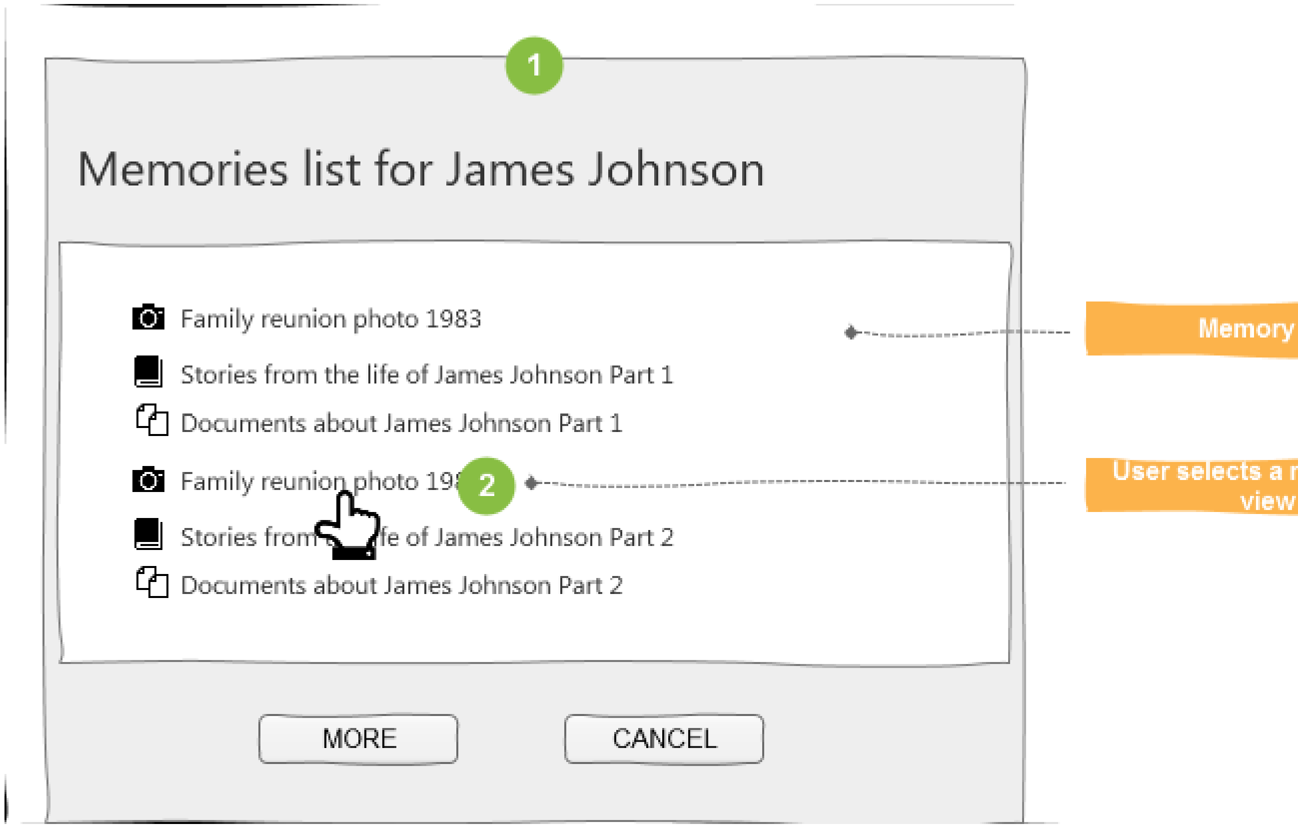
Programming Steps

1. Read the list of memories attached to a person, and display them. Optionally, read memories by type. Memory types include photos, documents, stories, and audios.
2. Allow the user to select a memory to view.
3. Read the memory details, and display them.

User Interface Suggestion

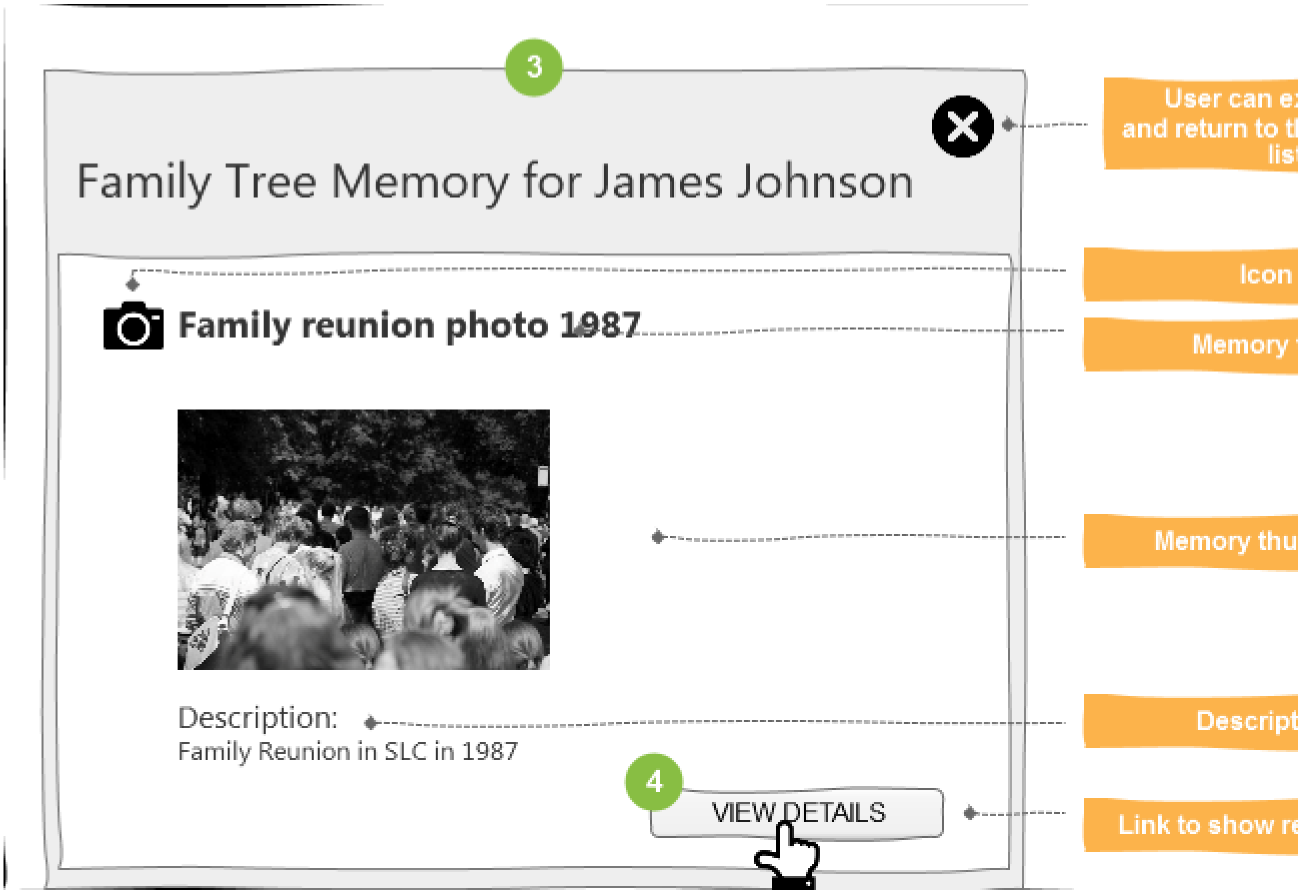
1. Show a list of memories for a person in the FamilySearch Family Tree.

2. The user selects a memory to view from the list.



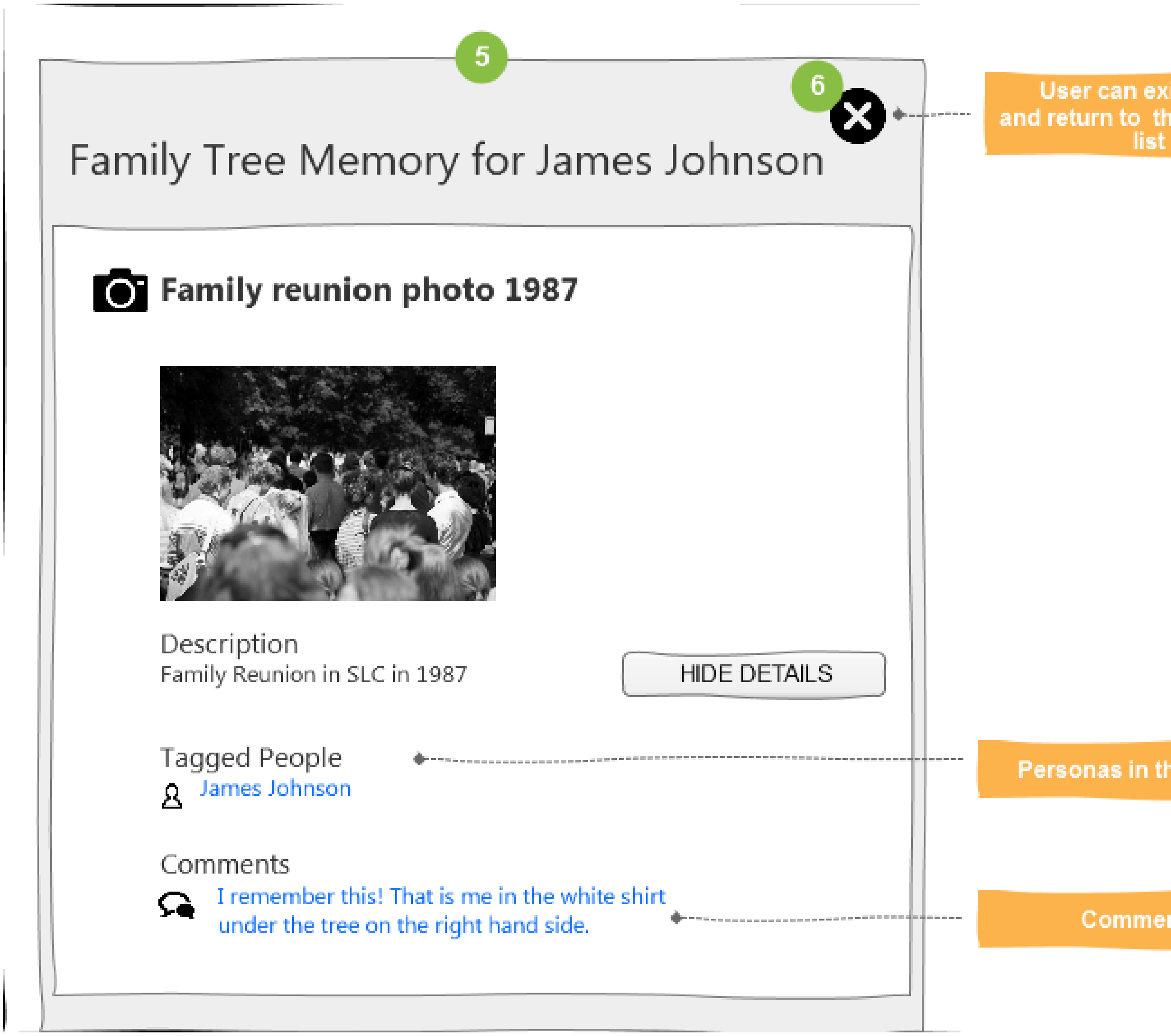
3. Display the following memory information:
    - Title.
    - A thumbnail, if available, with a link to view the memory.
    - Description.
    - An option to view the memory details.

4. The user selects the option to view details.



5. Identify the people referenced in the memory, and provide the option to view each person and to view comments on this memory.

6. Finish viewing, and return to the memories list.



Related Topics

- Read the list of memories
- Read the list of personas
- Read the comments on a memory

List Memories Personas

A person is identified in a memory by creating a persona. You can read a list of personas on a memory.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- The app has obtained the memory ID of a memory.

Programming Steps

1. Obtain the list of personas that are tagged on the targeted memory.
2. Display the personas.

User Interface Suggestions

1. Display a list of memories of a person. (a list could also be done for memories created by the current user.)
2. Allow the user to select a memory.
3. Display the list of personas.

Read Memories Contributed by the Authenticated User

Apps can allow users to perform the following tasks:

- View all of the memories the user created.
- Navigate through multiple memories, and select a memory to view the details.
- View the details of a memory, including:
    - Metadata (title, description, contributor, timestamp, and so forth).
    - Comments.

Assumptions

- The user is authenticated as a FamilySearch user.
- The user has previously contributed a memory.

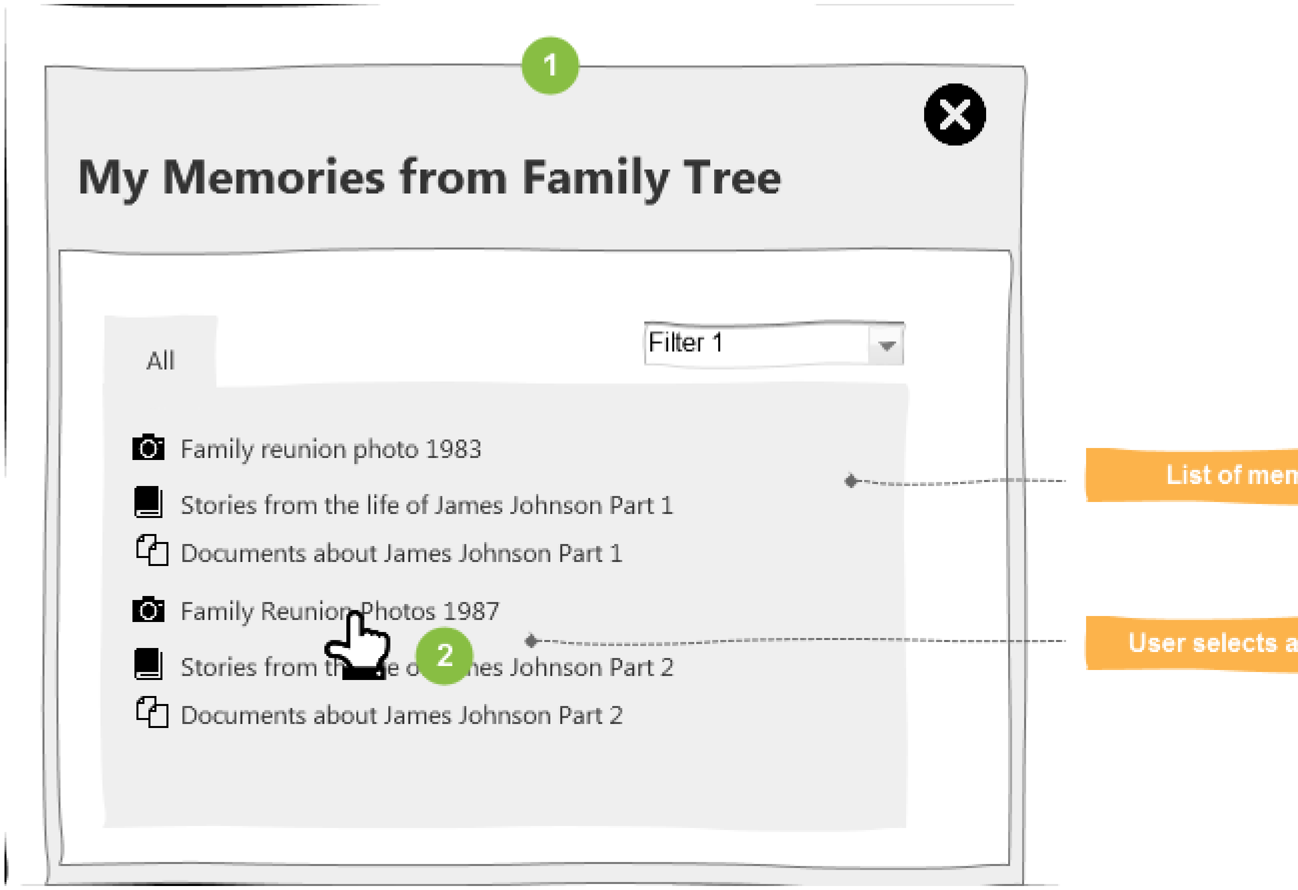
Programming Steps

1. Read the list of memories submitted by the current authenticated user, and display a list.
2. Allow the user to select a memory to view.
3. Read the memory selected by the user, and display it.

User Interface Suggestions

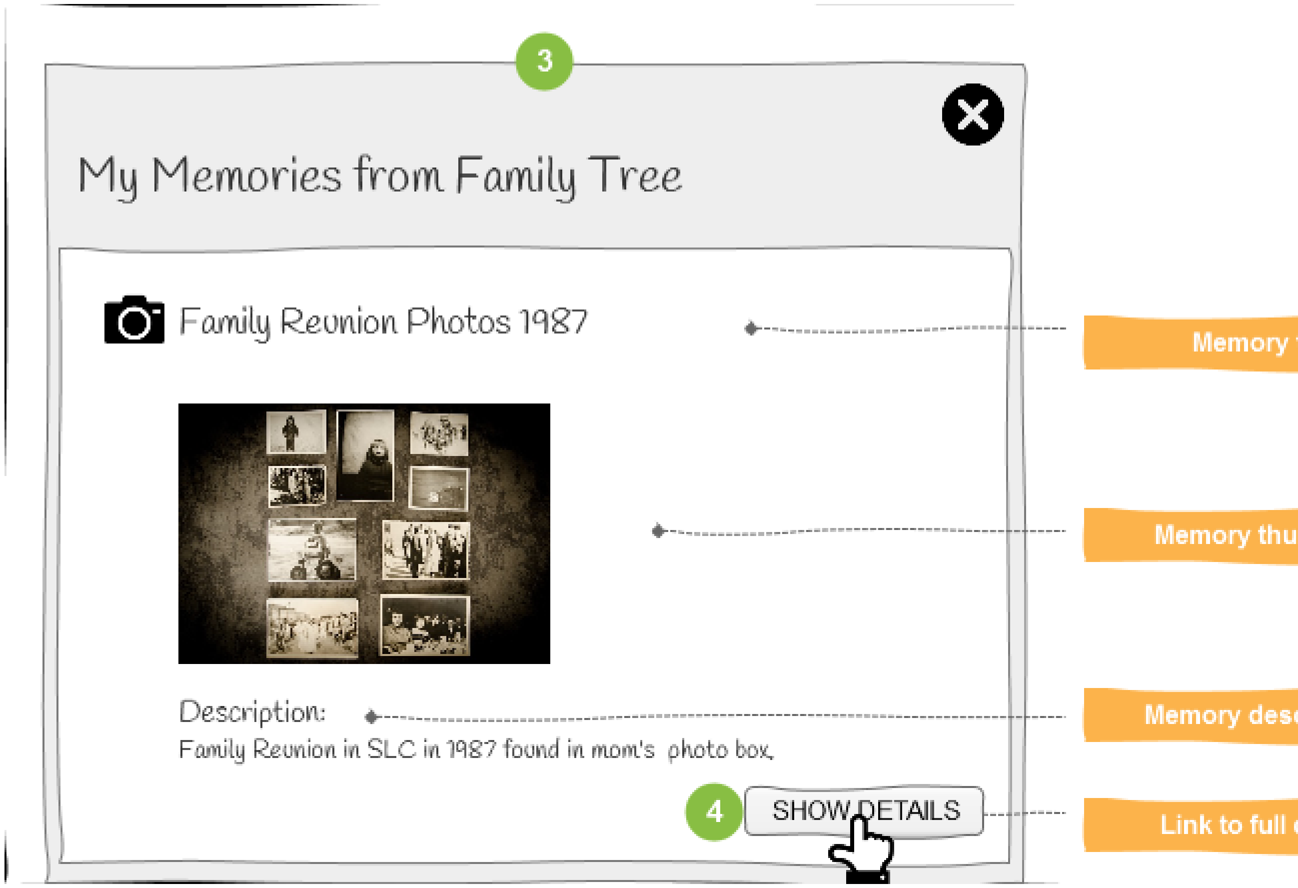
1. Display a list of the user's memories.

2. The user selects a memory from the list.



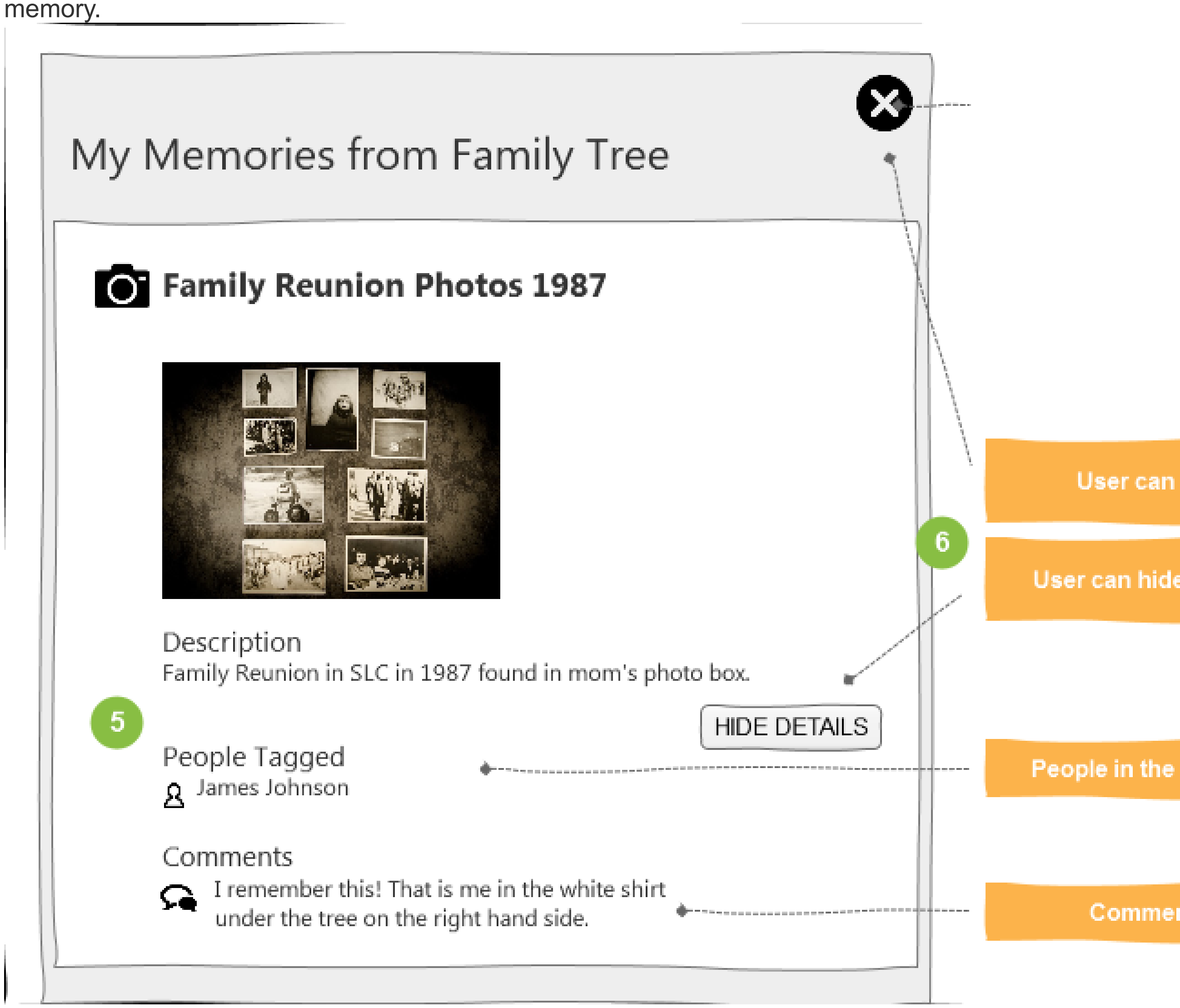
3. Display the following information:
    - Title.
    - A thumbnail, if available.
    - Description.
    - An option that can be selected to show the memory's full details.

4. The user selects the option to see additional information.



5. Display the following information according to the type of memory:
    - Full picture, story, or document.
    - People in the memory.
    - Comments.

6. The user can hide the details and exit viewing the memory.



Related Topics

- Read memory comments
- List Memory Personas to get a list of people in this memory.

Read Person Details and Relationships

Details can be read about a person and those who are related to that person such as spouses, parents, and children.

Assumptions

- You must have the person ID of the person you want to view.

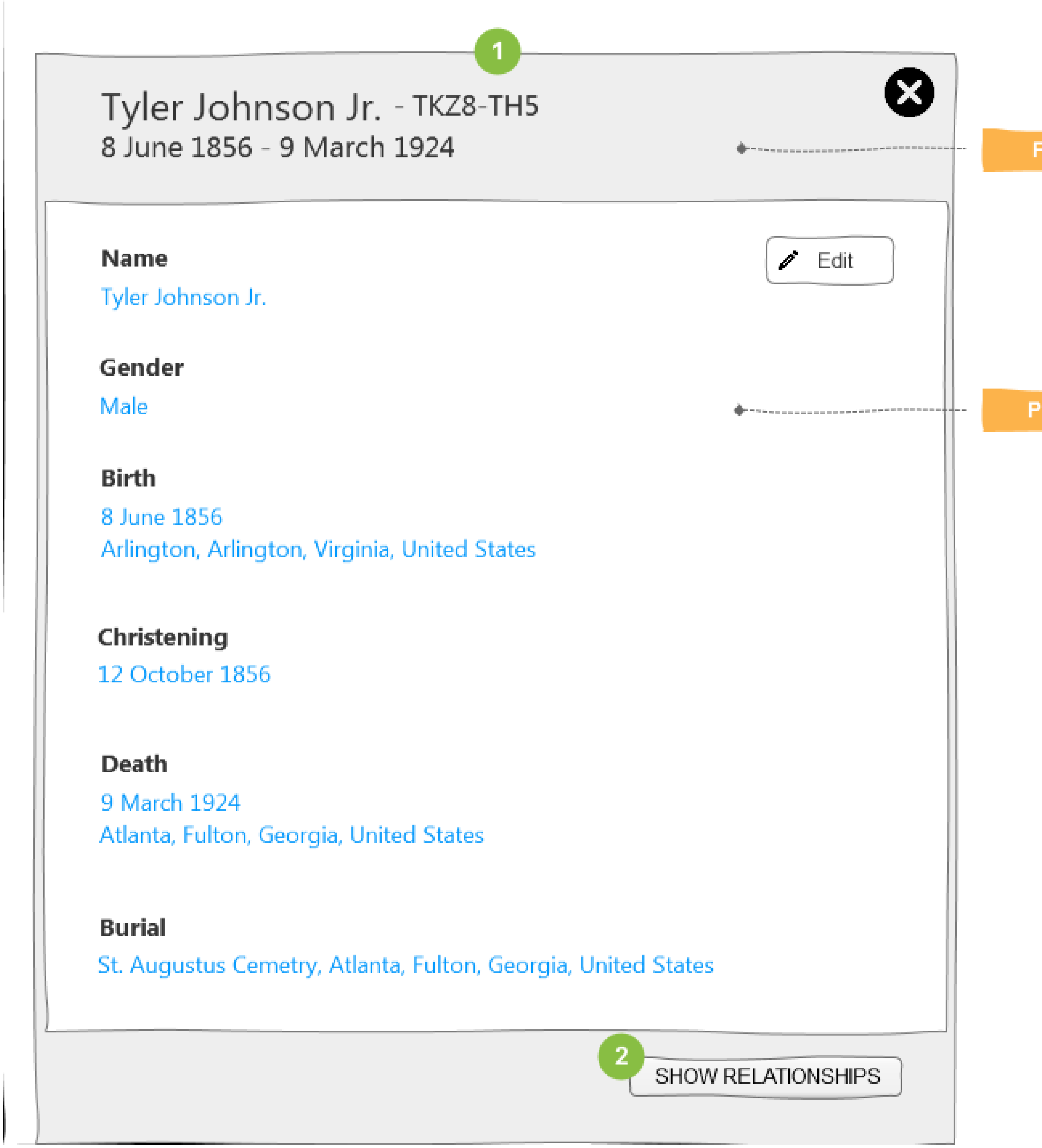
Programming Steps

1. Read a person and display the information.
2. Read spouse and display the information.
3. Read a parent and display the information.
4. Read child and display the information.

User Interface Suggestions

1. Show information about the person.
    - Name
    - Person ID
    - Sources that are tagged to the vital conclusions
    - Vitals include name, gender, and dates and places of birth, christening, burial, and death.

2. Offer the option to view relationships.



3. Show the following information about spouses, children, parents, and siblings.
    - Name
    - Event date
    - As applicable indicate relationship type: preferred spouse, preferred parents, biological, adopted, guardianship, foster, or step.

- Date entered, reason statement, contributor and the last modified date.

- Sources that are attached to the relationship.

Tyler Johnson Jr. - TKZ8-TH5
8 June 1856 - 9 March 1924

3

Family Members

+ Add Family Member   Edit

Spouses

Tyler Johnson Jr. - TKZ8-TH5
1856-1924

Molly Roberts - TKZ8-TJ9
1859-1929

Married
29 MAY 1878
Arlington, Virginia, United States

Children

George Tyler Johnson - TKY1-TP2
1880-1909

Timothy Jonathon Johnson - TLY7-
1883-1940

Sophia Johnson - TZH6-LP5
1885-1948

Buford Johnson - TPW9-KS1
1887-1955

Ruth Molly Johnson - TNG1-MV2
1890-1901

Parents

Tyler Johnson Sr. - TKA6-AH2
1825-1888

Mary Lois Stevens - RKD6-TN1
1835-1881

Married
2 MAY 1853
Oklahoma City, Oklahoma, Oklahoma, United States

Siblings

Sarah Mary Johnson - PTY5-HP2
1855-1910

Tyler Johnson Jr. - TKZ8-TH5
1856-1924

Jeanie Johnson - YGB8-FS2
1858-1928

John Dunford Johnson - TQT7-VS4
1860-1938

Matthew Johnson - XBH8-DJ1
1861-1939

Read and Start a Pedigree from Family Tree

The app reads, links, and copies person data for multiple generations from Family Tree beginning at a designated person. Starting a pedigree is sometimes known as:

- Starting a tree
- Copying a tree
- Importing a tree
- Seeding a tree

Start with the current user or another ancestor, and add generations from that person. The user should be allowed to select the scope of the copying or accept a reasonable default.

Assumptions

- The persons are being copied to a new third-party tree or extending a branch where there are no persons, so there is no need to check for matches or duplication between the FamilySearch tree and the third-party app tree.
- Any person that is having generations added must previously be linked to a FamilySearch Person ID.

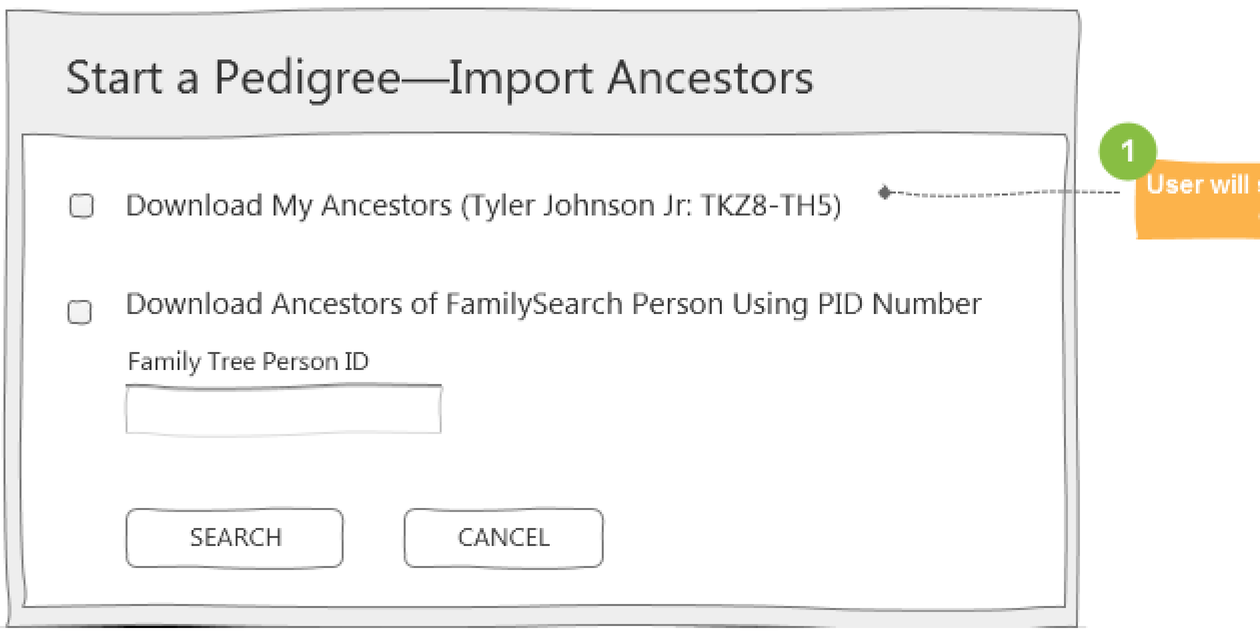
Programming Steps

1. Request an ancestry for a number of generations for either the signed-in user or a selected person ID.
2. Parse and display the person data and relationships with a chart or a tree.

User Interface Suggestions

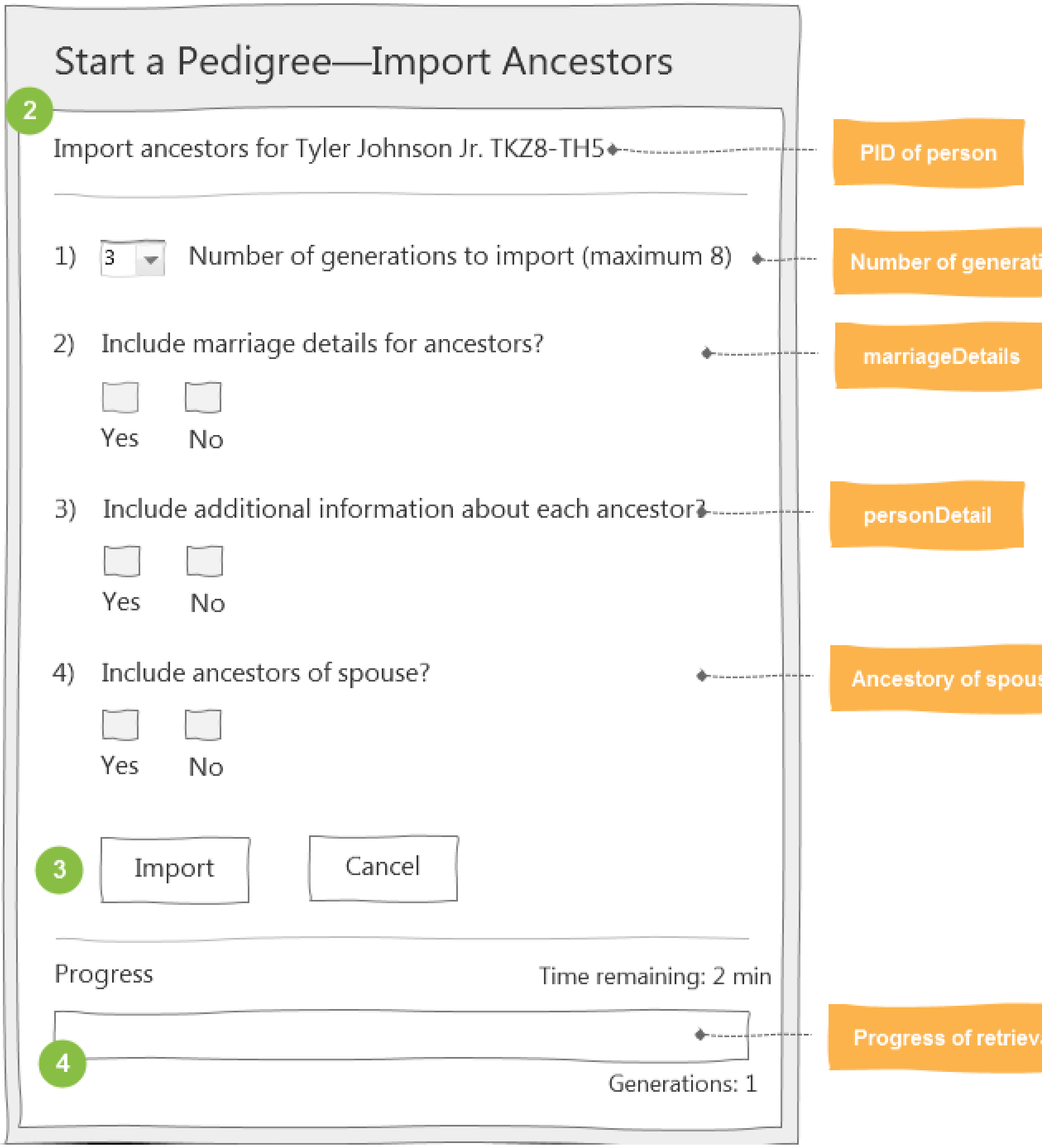
1. Display a choice to add generations to the signed-in user or a linked person that is lacking

ancestors.



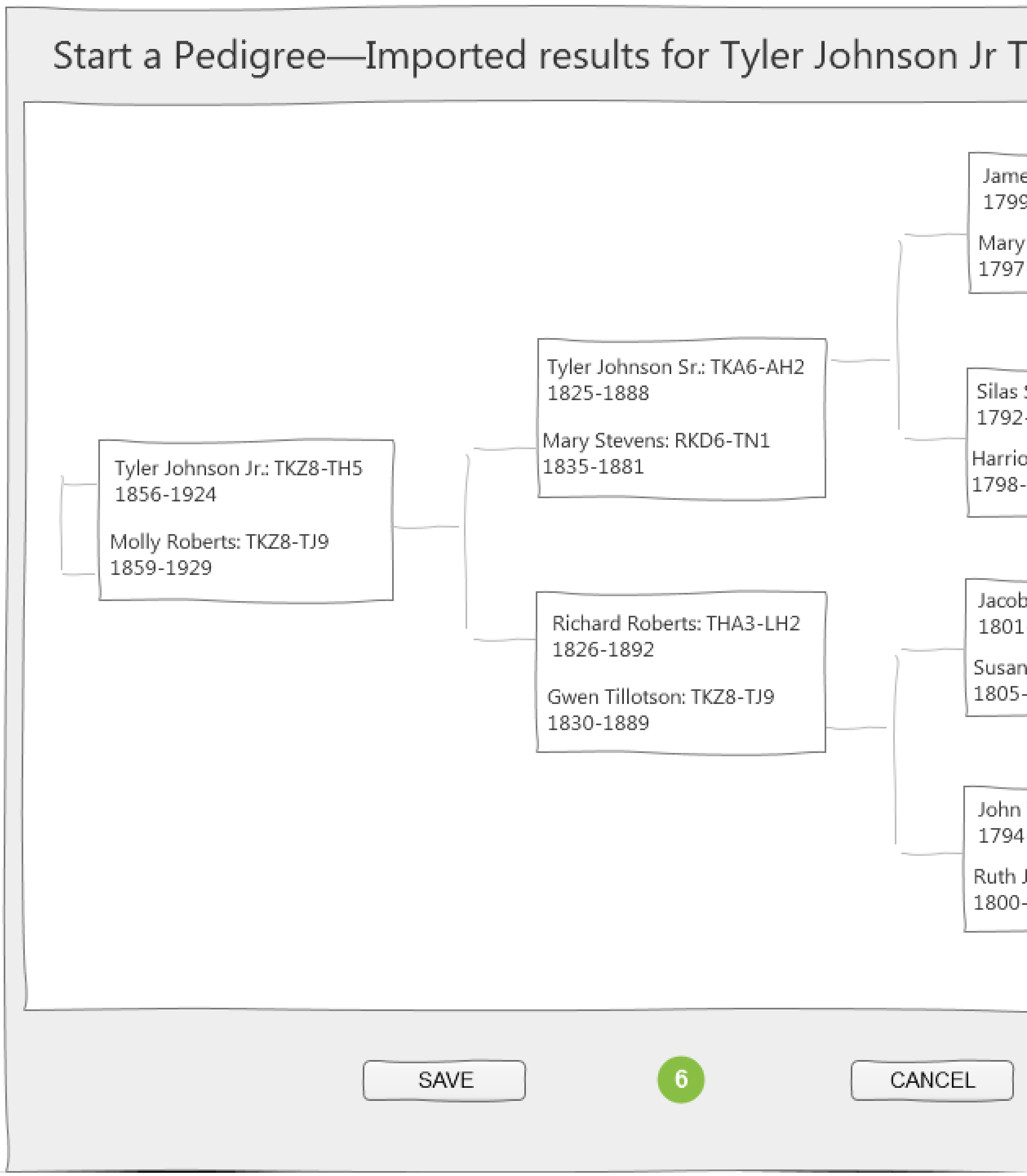
2. Display the following parameters that can be entered to set the criteria for the download.
    - ***generations*** The number of generations being queried. Default of 4 and maximum of 8.
    - ***marriageDetails*** Yes or No selection to include the marriage details for each person that is being downloaded.
    - ***person*** The PID of the person whose ancestry is to be read. The default is the signed-in person.
    - ***personDetail*** Additional person details are downloaded for each person in the ancestry.
    - ***spouse*** This entry also provides the ancestry of the spouse.
3. Ask the user to confirm or change the above parameters.
4. Upon confirmation, initiate the read request, and show a progress bar or other indicator to communicate the progress and approximate time

remaining.



5. Display the results of the query in report form or a pedigree.

6. SAVE or CANCEL the downloaded information.



Read Person Portrait

Users can access a FamilySearch Family Tree person portrait by requesting the thumbnail (Portrait), multiple sizes (Portraits), or the thumbnail and an URL address with a default image returned (Portrait with Default) when there is no portrait image.

Assumptions

The user is signed into the third-party application and knows the person ID of a FamilySearch Family Tree person. The Person ID can be obtained from a previously matched and linked person or by requesting it from the user.

Programming Steps

Request one of the following:

- Read Person Portrait
- Read Person Portrait with Default
- Read Person Portraits

User Interface Suggestions

1. The person portrait image can be shown on a
    - Person Summary Page
    - Person Detail Page
    - Pedigree or another family chart page

Person Search and View

An app can allow users to:

- Search for a person in FamilySearch Family Tree by entering parameters. This is done in order to obtain the person ID which is needed to perform tasks such as:
    - Displaying a person's information.
    - Selecting the starting person of a pedigree.
    - Attaching a source, memory, or artifact to a selected person.
- Search the FamilySearch Family Tree for possible duplicates of a person based on selected criteria. This is done in order to merge or delete duplicates.
- Search the FamilySearch Family Tree for possible matches of a person not yet in the Family Tree. This is done in order to associate a person in a third-party application with a person in the FamilySearch Family Tree.

Assumptions

- The user has authenticated with FamilySearch.

- If using the JavaScript SDK functions, you must have access to the JavaScript SDK.

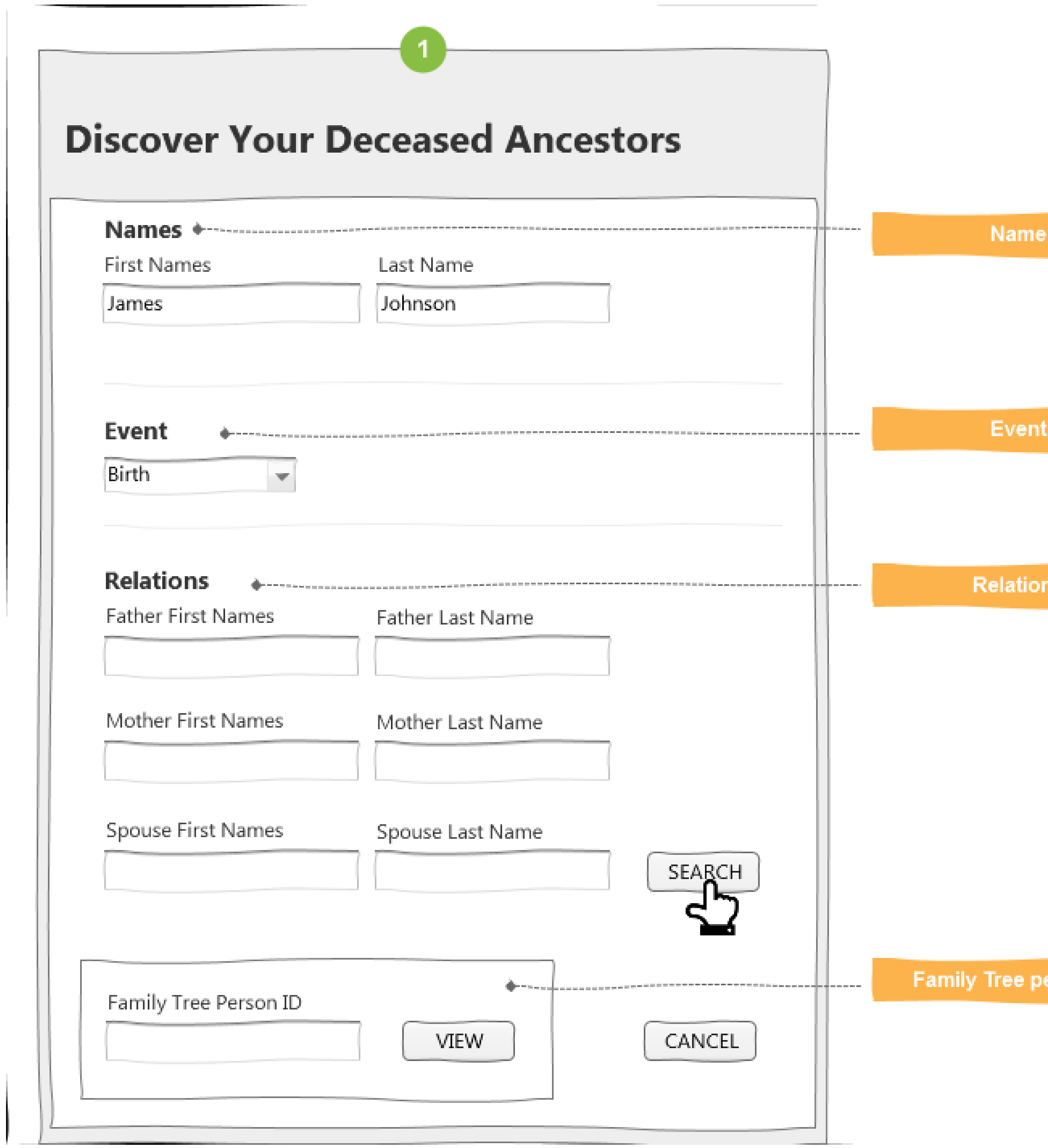
Programming Steps

1. Search by parameters to find a Family Tree person.
2. Read a Family Tree person that you found.
3. Search for duplicates within FamilySearch Family Tree.
4. Search for possible matches to a person in Family Tree.

User Interface Suggestions

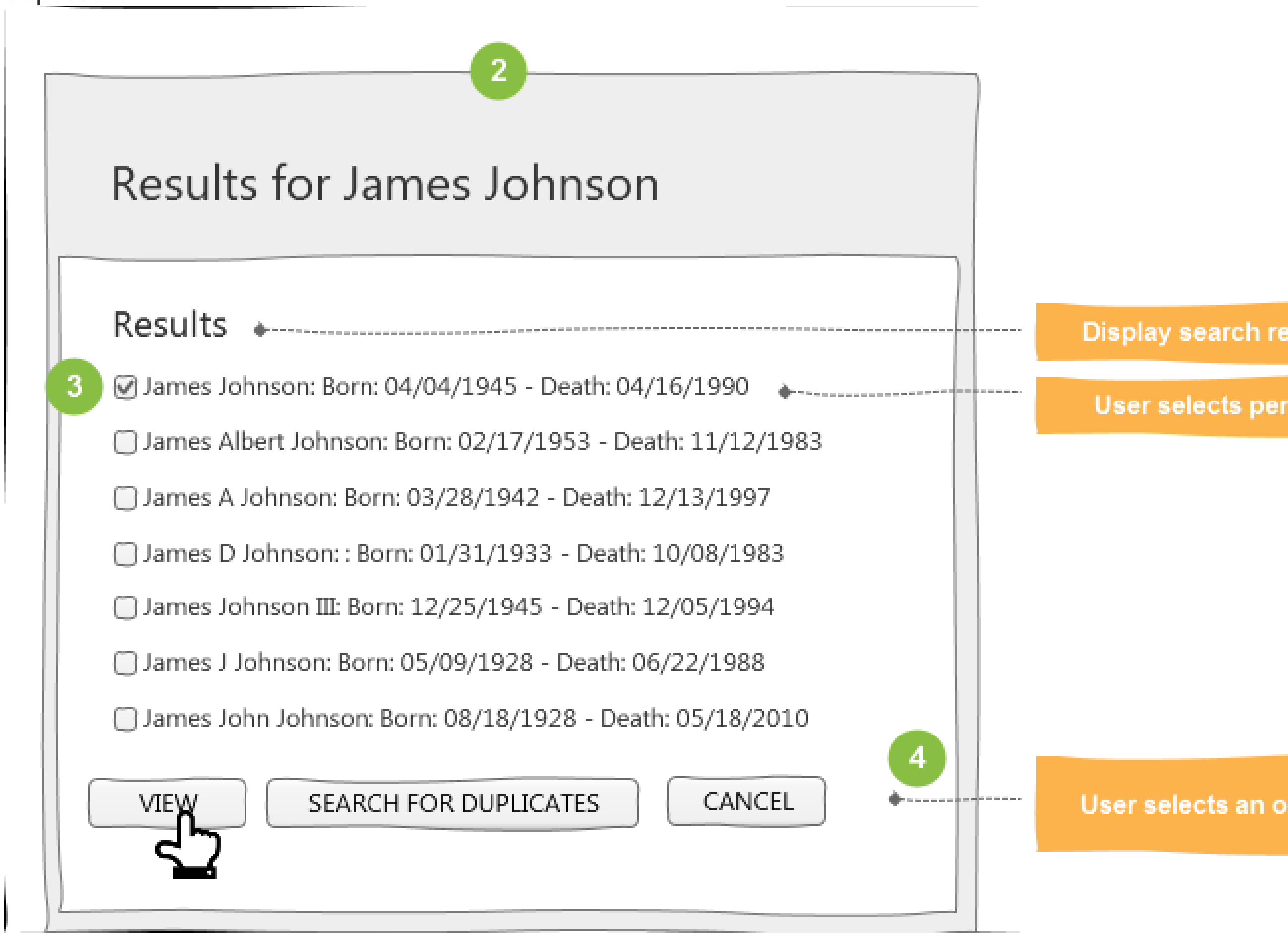
This shows how you could search for a person in the FamilySearch Family Tree, and check for duplicates.

1. Enter person search parameters (name, event, relations), then SEARCH, or enter the Family Tree person ID to VIEW a person.



2. Display the search results for review.
3. The user selects the person that they are looking for.
4. The user chooses to VIEW the person information, SEARCH FOR DUPLICATES, or CANCEL. ***NOTE:*** If you have an application tree person that you want to match with a Family Tree, you would search for possible matches instead of search for

duplicates.



User Interface Suggestions for JavaScript SDK Apps

Note: If a FamilySearch access token has not been previously obtained at the time an SDK function is called, the SDK function will take necessary steps to obtain one in order to complete the function.

1. Obtain the actual Family Tree Person ID.
2. Call the getPerson SDK function.
   The person data is returned.
3. Collect desired search parameters (name, birth date, etc.) of a person from the user.
4. Call the getPersonSearch SDK function.
   The search results are returned.
5. Display the search results and ask the user to select one.
   The user selects the desired person.
6. Call the getPersonMatchesQuery SDK function.
   The search results are returned.

7. Display the search results and ask the user to select one.
   The user selects the desired person.

Read and View Sources Contributed by the Authenticated User

Authenticated users can read sources they stored along with attribution information. Attribution if available consists of the following information.

- Contributor
- Last modified date
- Reason statement

Assumptions

- The user has previously contributed a source.
- You know the Person ID of the person whose sources you want to read.

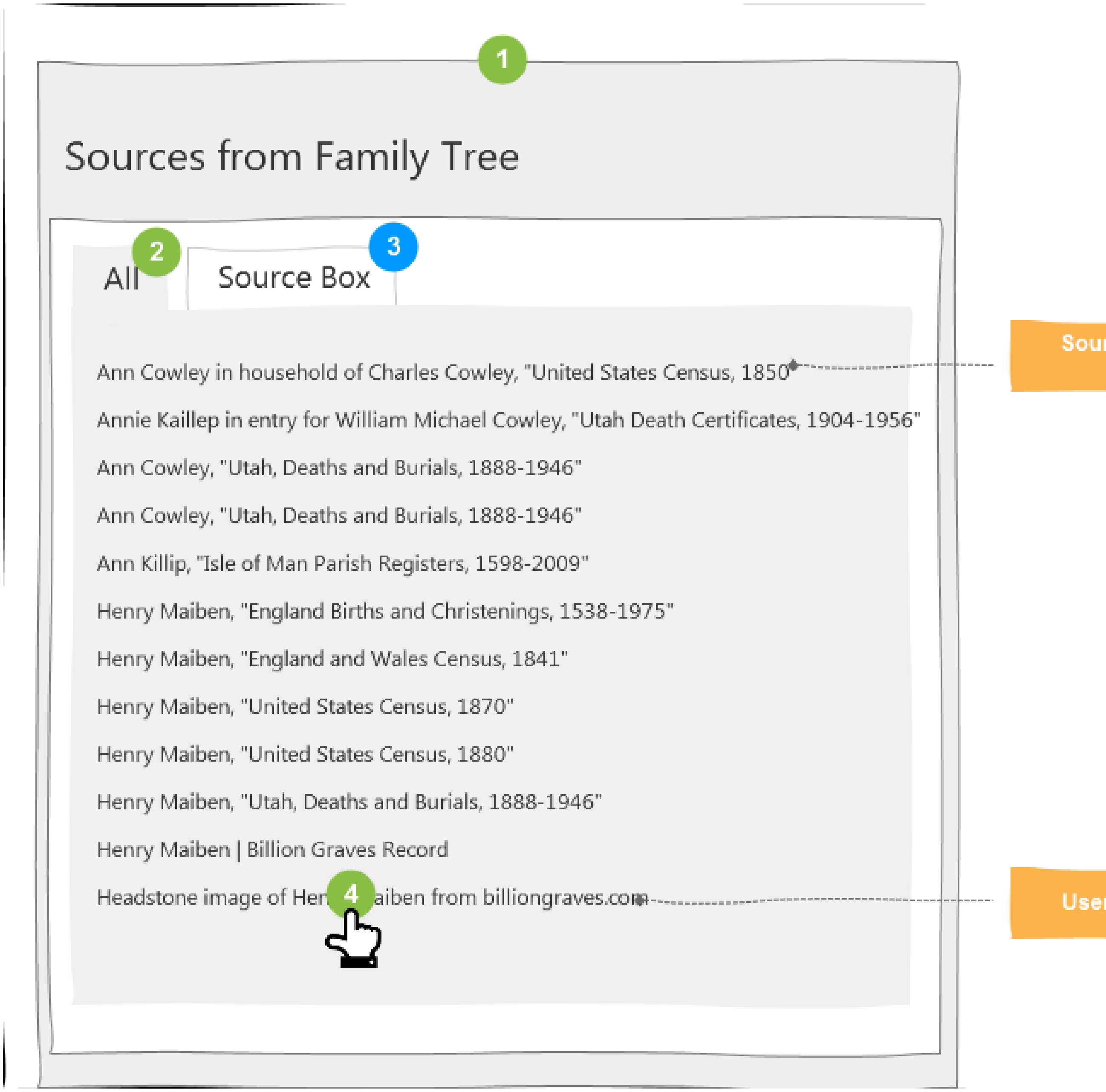
Programming Steps

1. (Optional) Read the source box collection of the current user, and display the results.
2. Read the sources attached to a person, and display them.
3. Read source details of a source, and display them.

User Interface Suggestions

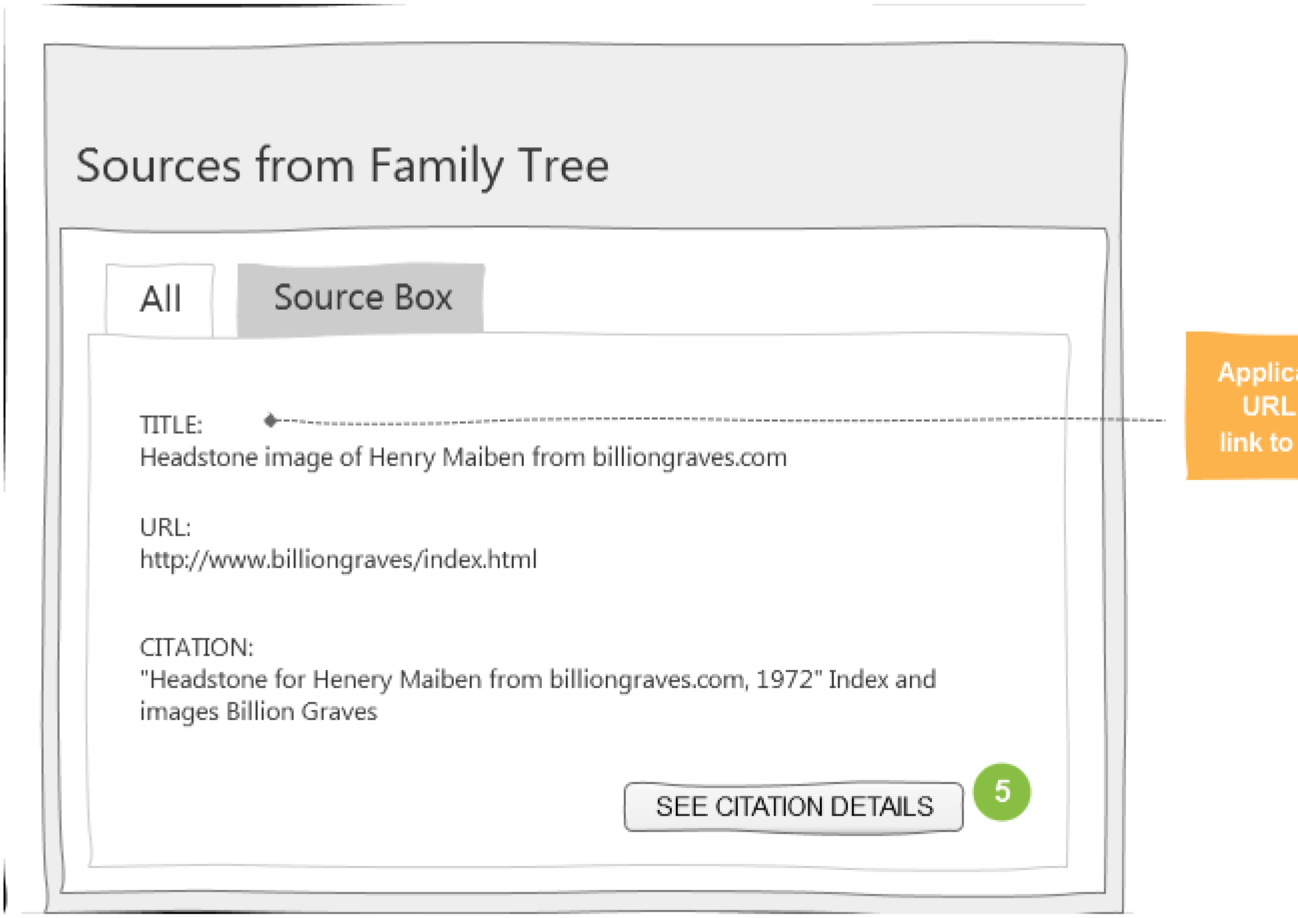
1. Request to see the sources attached to a person.
2. Display the list of the sources.
   The user reviews the list of sources.
3. (Optional) Read the source box collection, and display the list of sources.

4. The user reviews a list of sources, and selects one of the items in the source list.



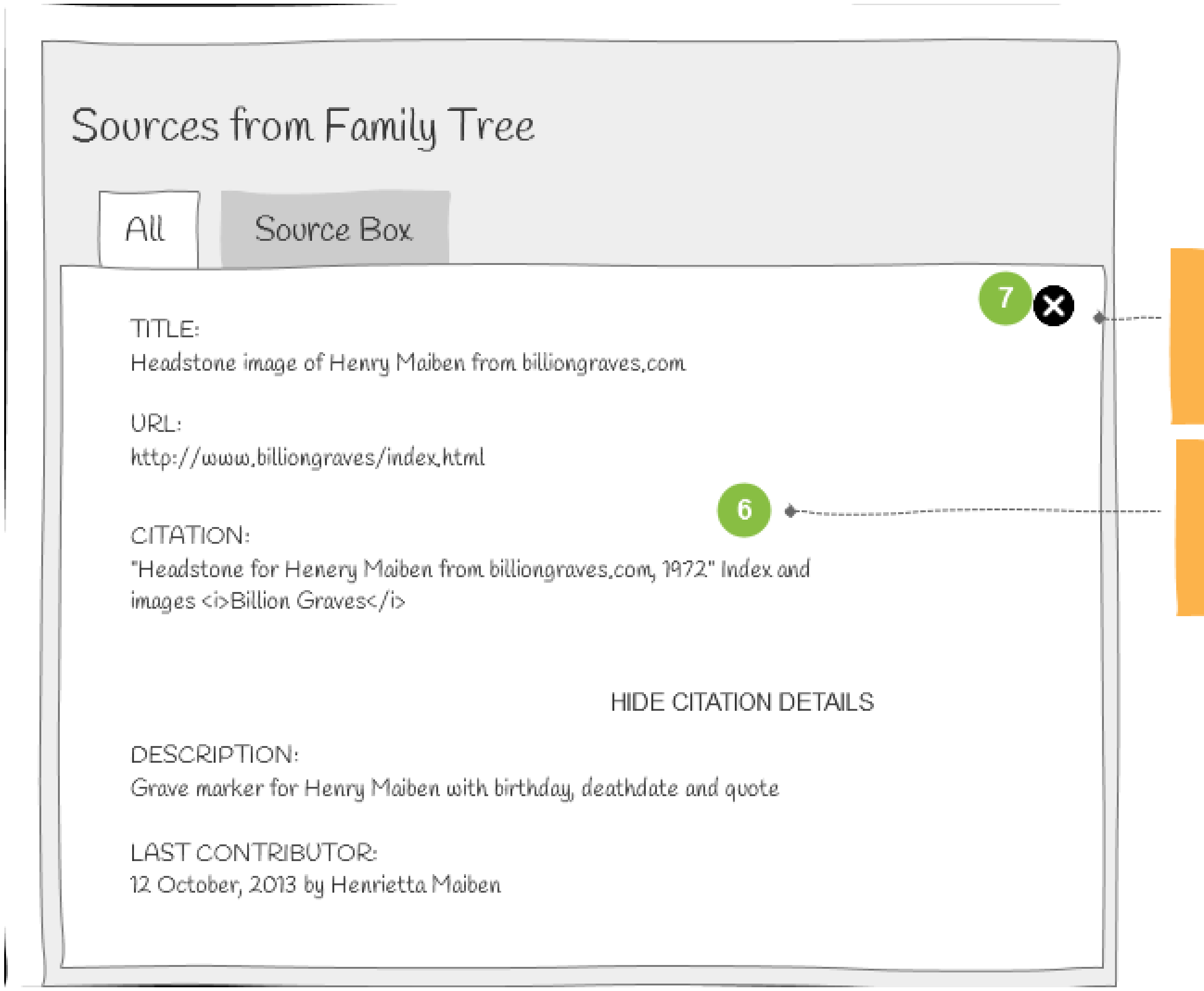
5. Read the selected source, and display the title, URL, citation (partial), and provide an option to see the full details of the

citation.



6. Allow the user to see additional information that includes the full citation, notes, and full attribution that includes last contributor, date, and reasons (if available).

7. Allow the user to exit and return to the sources list.



Write Memories Implementation Guide

Create Memories (Not Attached to a Tree Person)

Users can create memories that are not attached to a person in the Family Tree.

Assumptions

- The user is signed-in to FamilySearch through the third-party application.
- The user has selected the option to create a memory.

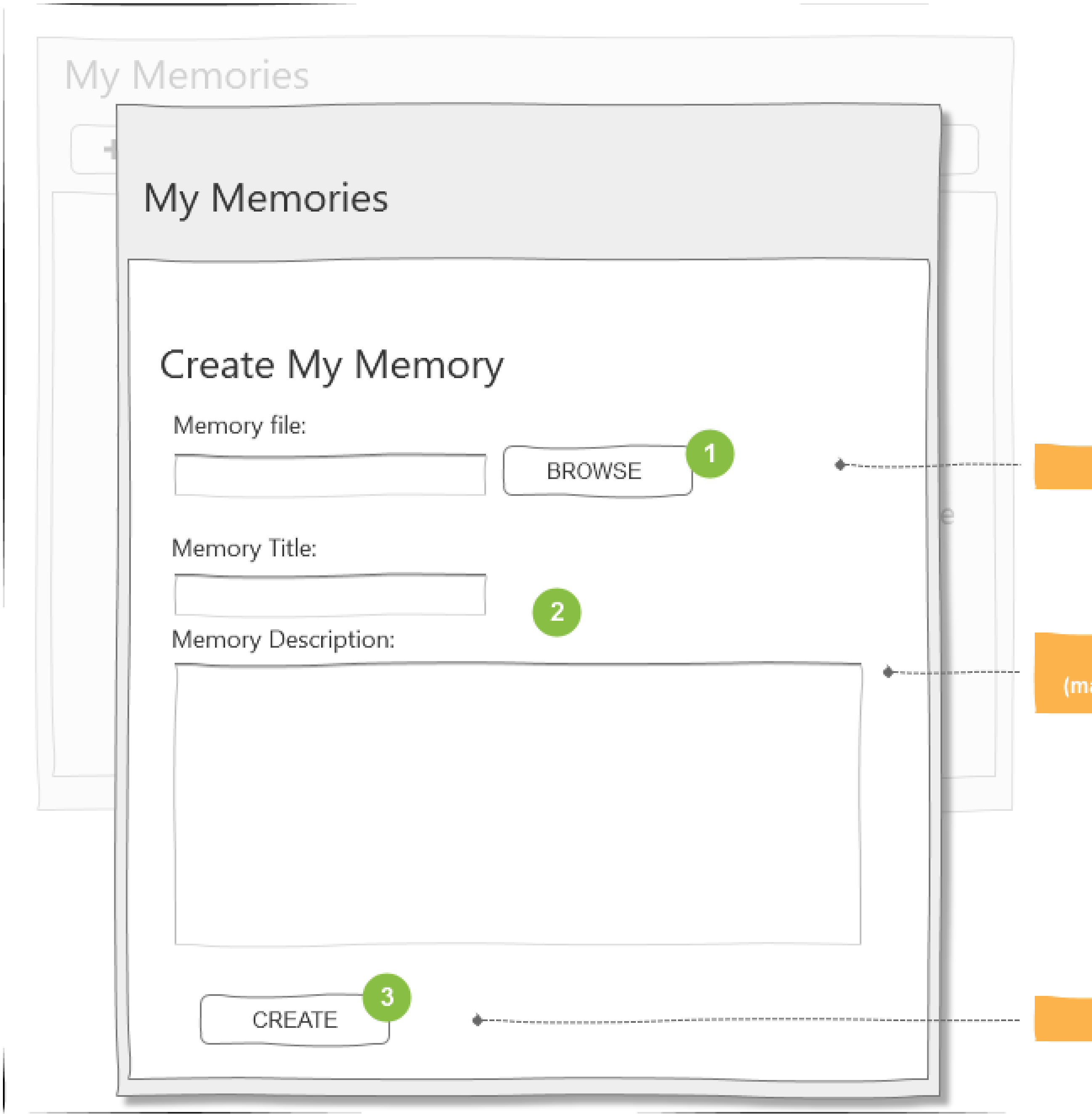
Programming Steps

1. Allow the user to select a file to store as the memory. This could be image file, a PDF, a text file, or an audio file.
2. Allow the user to input title and description of the memory.
3. Save the file as a memory based on the file format. (image,PDF, text, or audio)

User Interface Suggestions

1. The user selects a file to upload as a memory. The file could be an image, a PDF, a text, or an audio file.
2. The user adds a title and description for the memory.

3. The user clicks **Create**, and the app stores the memory based on the file format.



Create Memories with a Person Attached

Users can create a memory for a person in the tree by uploading a file. This action creates a memory and attaches it to a person in one request.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- The user has already identified the person in Family Tree to attach the memory to.

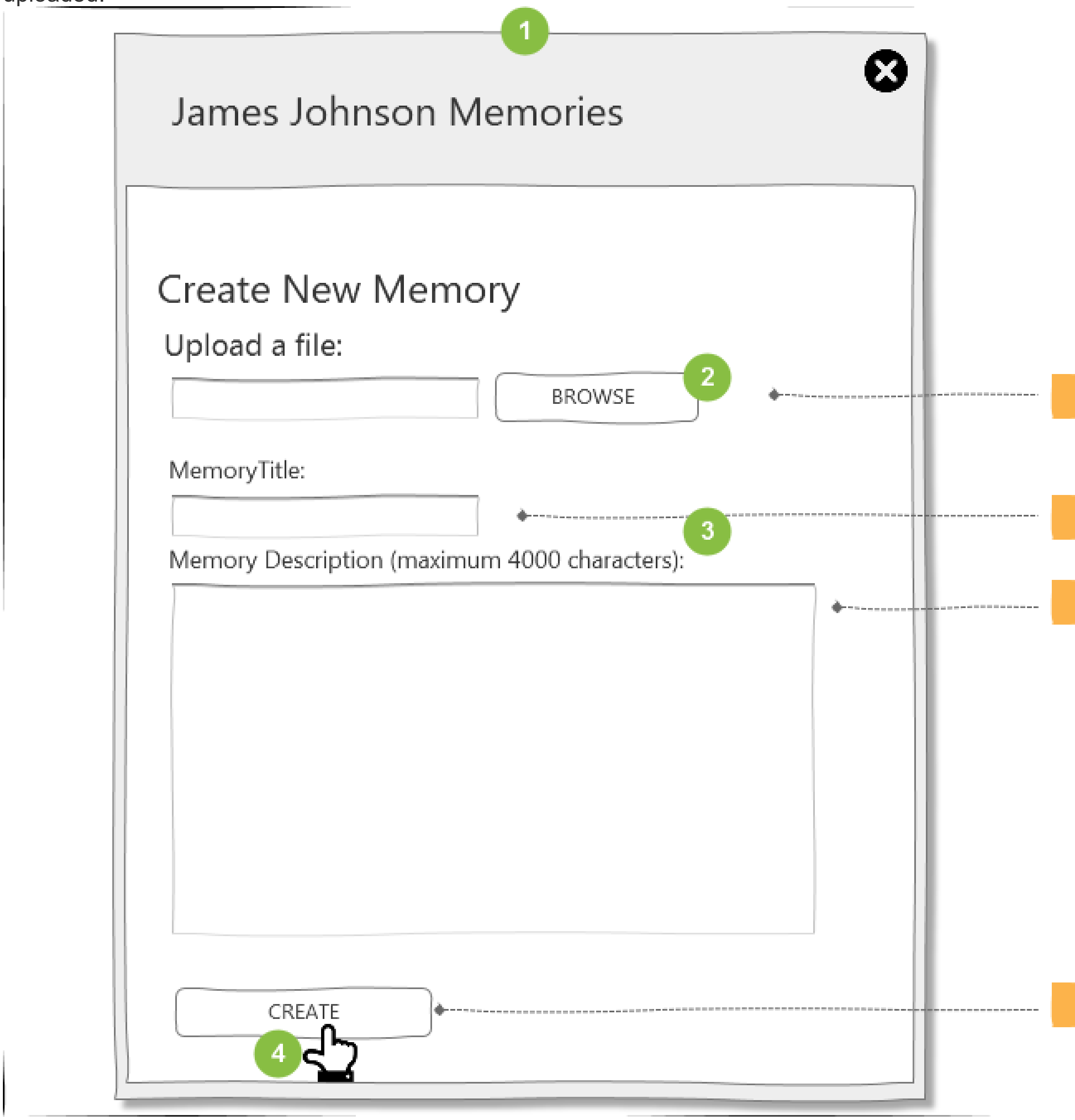
Programming Steps

1. Allow the user to select a memory file to upload. The file could be an image, PDF, text, or audio file.
2. Allow the user to input the title and description of the memory.
3. Create the memory (image, PDF, text, or audio).

User Interface Suggestions

1. The user selects the option to add a memory to a known person in Family Tree.
2. The user selects a file to upload as the memory.
3. The user adds a title and description for the memory.

4. The user creates the type of memory based on the file format. The file is uploaded.



Memories Upload Agreement

Users are required to accept the Memories Upload Agreement the first time they upload a photo, document, or audio file to FamilySearch.org.

Assumptions

- The user must be signed in to FamilySearch.org through the third-party application.
- The application can select a photo, document, or audio file to upload.

Programming Steps

1. Display the Memories Upload Agreement the first time the user tries to upload a photo, document, or audio file to FamilySearch.
2. The words "FamilySearch Content Submission Agreement" in the text, should be linked to https://familysearch.org/legal/familysearch-content-submission-agreement
3. The user must mark the checkbox to accept the statement "I have read and agree to the Submission Agreement." The words "Submission Agreement" should also be linked to https://familysearch.org/legal/familysearch-content-submission-agreement.

Optional

If the application does not have photo, document, and audio guidelines substantially similar to the following three guidelines, the application should provide the user with the FamilySearch guidelines:

- Photos Guidelines.
- Documents Guidelines.
- Audio Recordings Guidelines.

Memories Upload Agreement

By continuing with the upload process, you confirm that you have the right or permission to share any images you submit, and you agree to the terms and conditions of the FamilySearch Content Submission Agreement. You also acknowledge that any images you upload become part of the collection hosted by FamilySearch.org, which is publicly viewable and accessible by anyone online. You will be able to remove images you have contributed to FamilySearch.org, but FamilySearch.org is under no obligation to monitor or inhibit the use of contributed images by others.

[ ] I have read and agree to the Submission Agreement.

Edit Memories Starting from a Memory List

Users can edit memories, starting from a list of the memories that the user created.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.

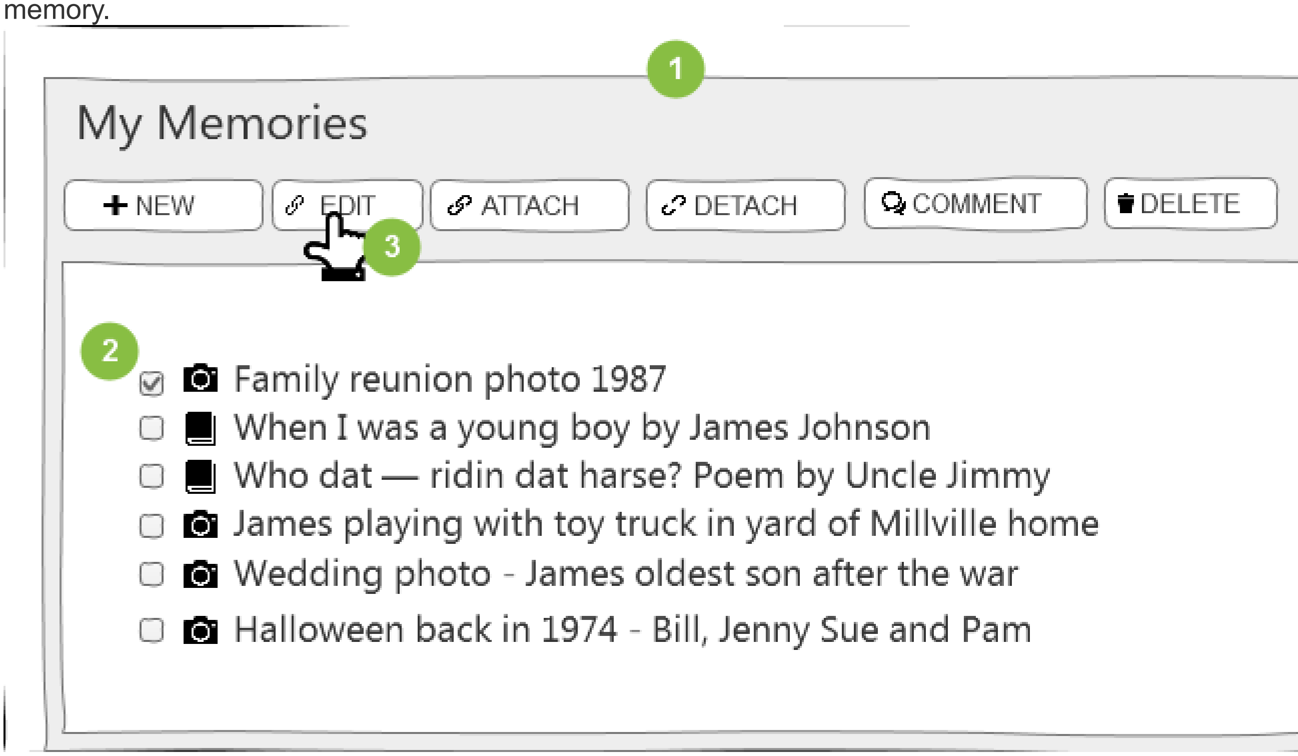
Programming Steps

1. Display memories that the user created.
2. Enable the user to edit a selected memory.

3. Save the changes.
   Note: See Update Story for changing text of a story artifact.

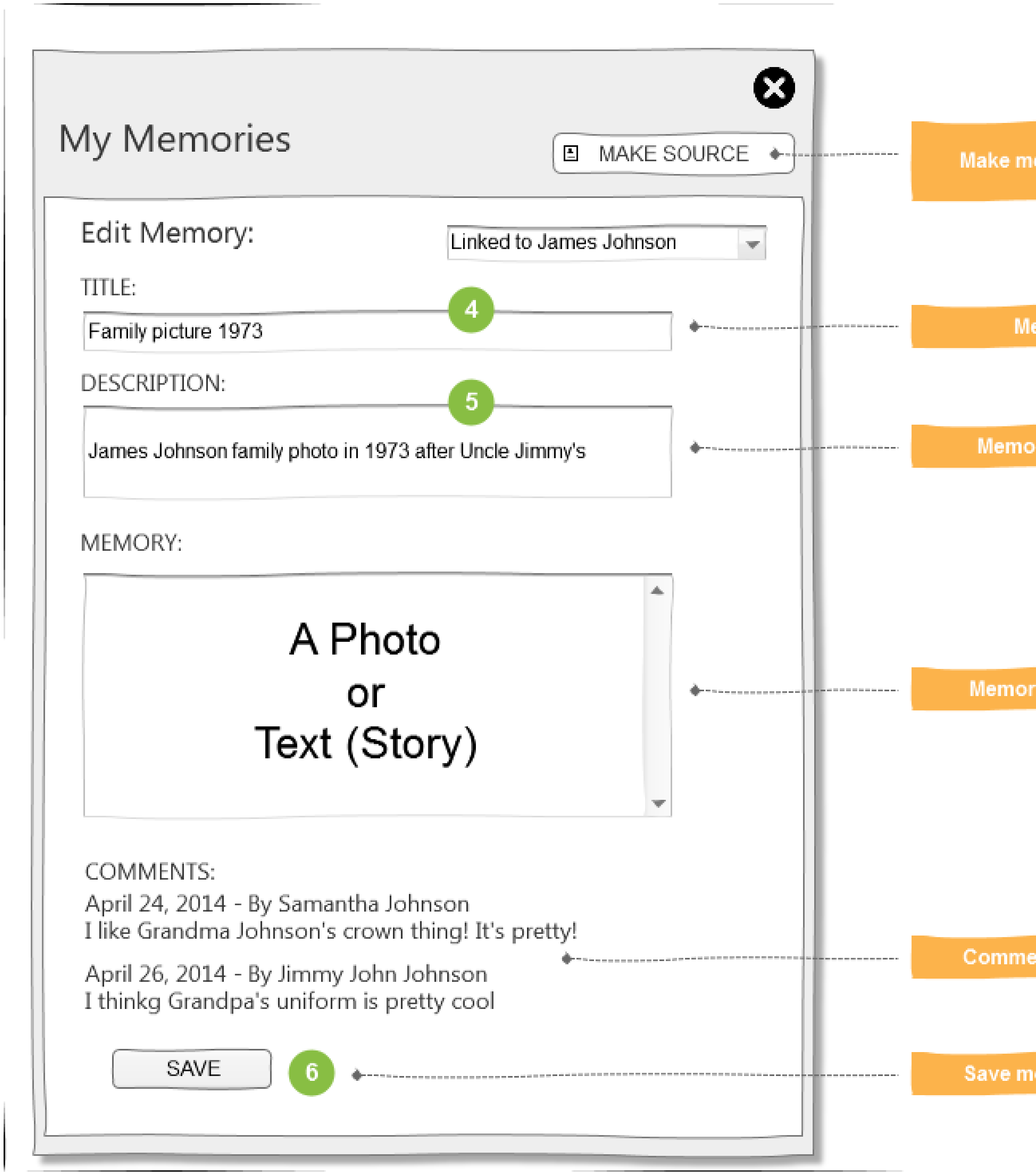
User Interface Suggestion

1. Display all of the memories that the current user has created.
2. Allow the user to select a memory.
3. The user selects the option to edit the memory.



4. Allow the user to edit the title.
5. Allow the user to edit the description.

6. Save the edits.



Additional References

[Read memories](#)

Create Personas to Attach Memories to Tree Persons

When you create a persona, you identify a person in a memory (tag the person) which associates (attaches) that persona with a Family Tree person.

Note: A floating memory is a memory that was created but was not attached to a person in the tree. This process is used to attach floating memories to people in Family Tree. Memories that are already attached to a person in the tree may also include other people who are tagged and attached to those people in Family Tree.

Assumptions

- The user is signed in to FamilySearch.org through the third-party app.
- The user has selected a memory that he or she wants to attach to a person.

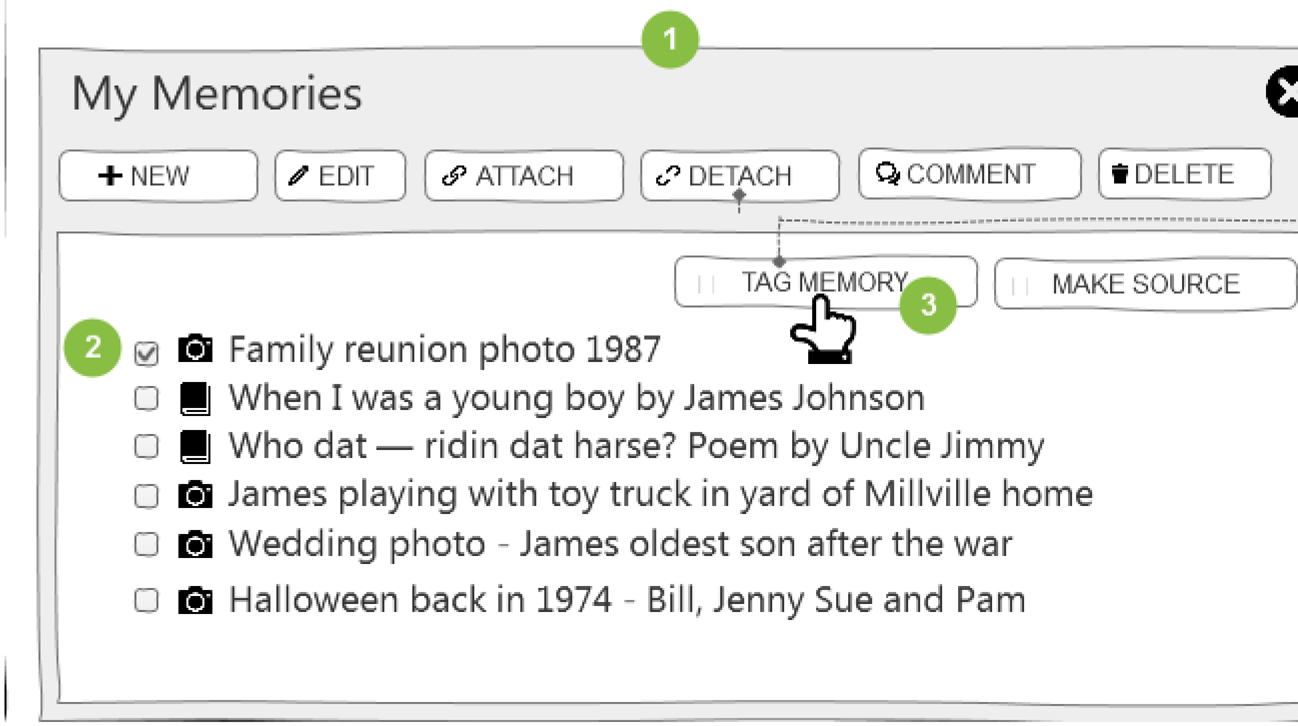
Programming Steps

1. Display the memory artifact, and collect tag information.
2. Locate the person ID of the person to attach to in Family Tree.
3. Create the persona.

User Interface Suggestions

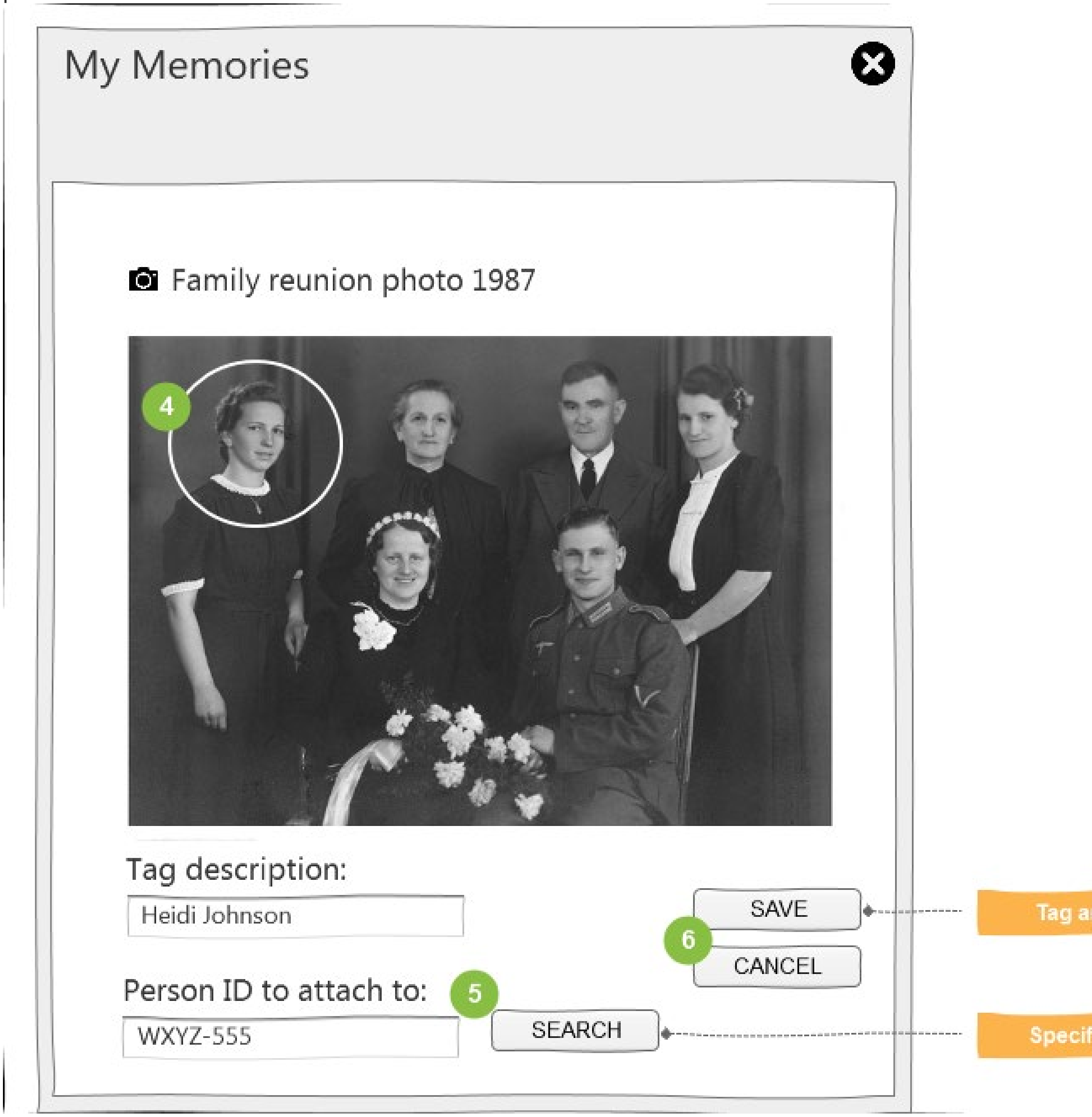
1. The app displays all of the memories that the user created.
2. The user selects a memory to attach.

3. The user selects the option to tag the memory.



4. Tag a memory.
   - For an image:
     - The user selects the portion of the image that will be associated with the tag.
     - The user enters the name of the person who is represented in the selected portion of the image.
   - For a document, a story, or an audio, the user enters the name of a person who is in the document.
   - **Notes:**
     - Use a square tag to create a round portrait. The user owns the tag, and no one else will be able to fix a poorly placed tag.
     - For an imaged-based document, tags can be rectangular and fit the document.
     - No portraits are created from document tags.
     - For audios, PDFs, and stories, no rectangular region is supported.
5. The user enters the person ID of the person in Family Tree to attach the memory to. Use the **Search** button to find the person.

6. The user clicks Save, and the app creates the persona.



The memory persona is created and attached to the person in Family Tree.

See Also

- [Memory persona]()

Comment on Memories Starting from a Memory List

Comments can be added to memories.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- The app has provided a way for the user to select a memory to comment on.

Programming Steps

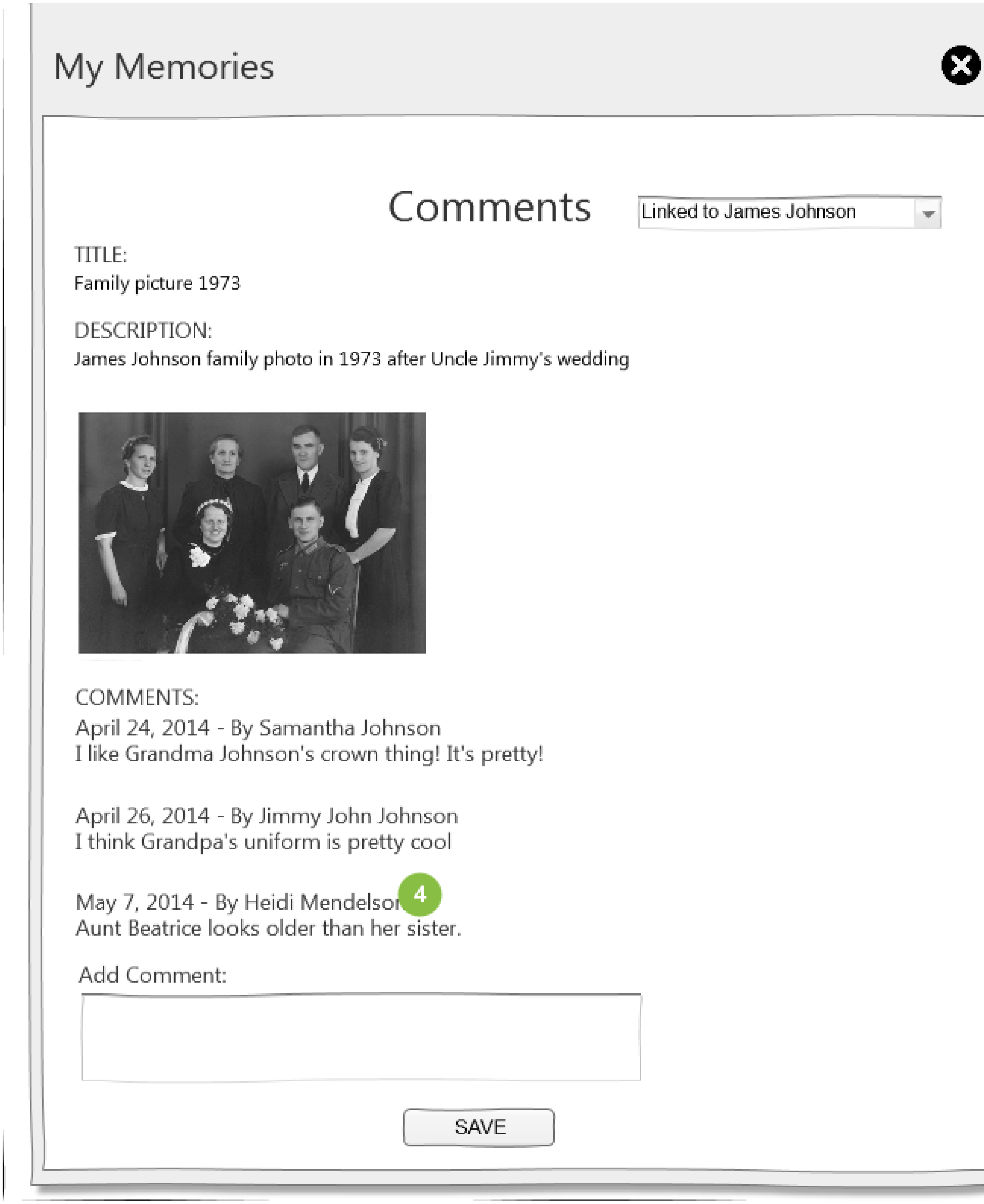
1. Identify a memory that you want to comment on.
2. Input a comment, and save it.

User Interface Suggestions

1. Display a memory and associated comments, with an option to add a comment.
2. The user enters the comment.

3. The user posts the comment.

4. The new comment shows with other comments.



Related Topics

- Obtain a list of memories contributed by the current user
- Read a selected memory
- Read comments on a memory

Create a Source from a Memory

You can designate a memory as a source reference by creating a source description for the memory.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- A memory already exists that the user wants to identify as a source.

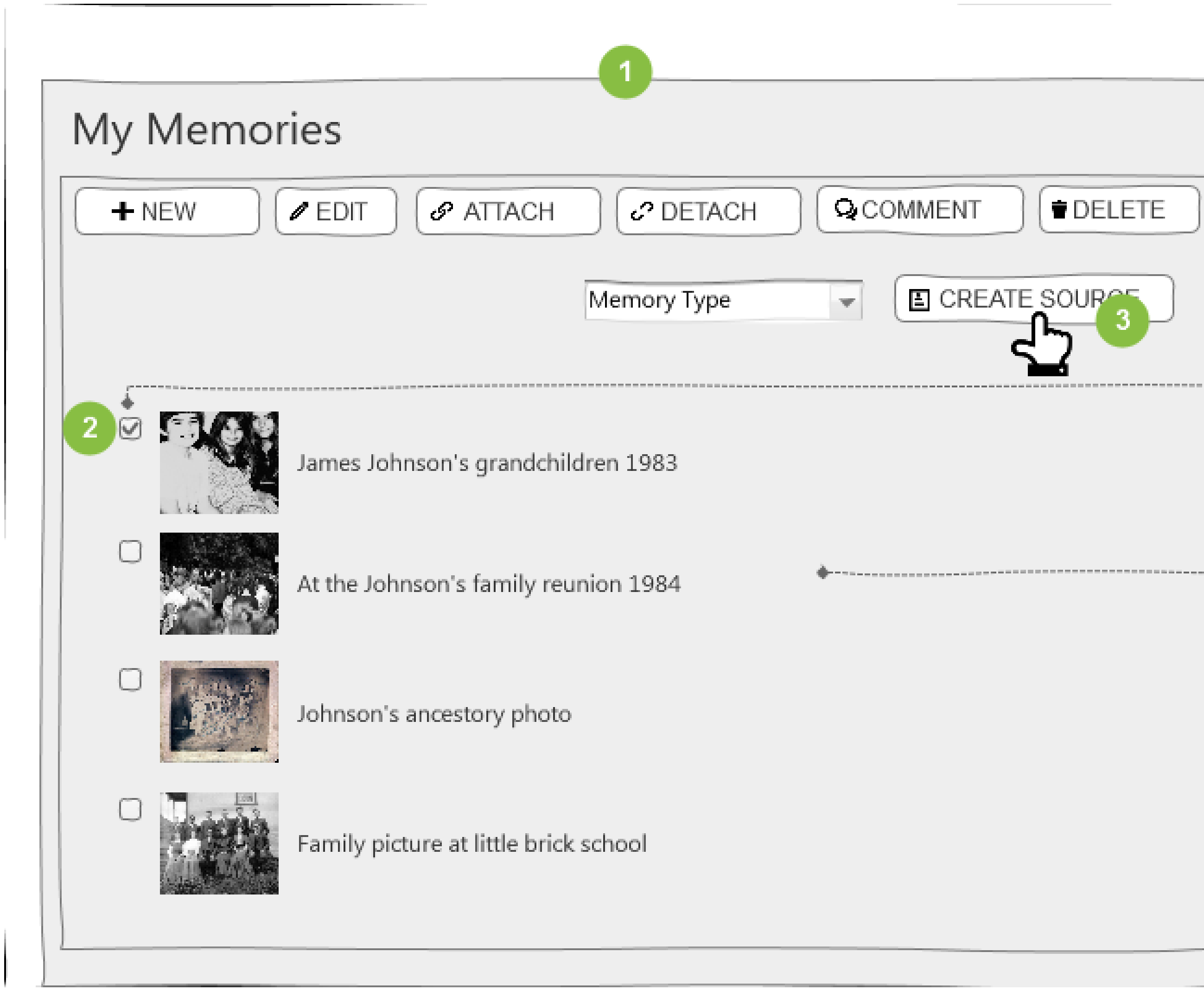
Programming Steps

1. Display a list of memories.
2. Have the user select a memory and provide a source title and description.
3. Identify the person ID the source is for.
4. Create a source out of the existing memory.

User Interface Suggestions

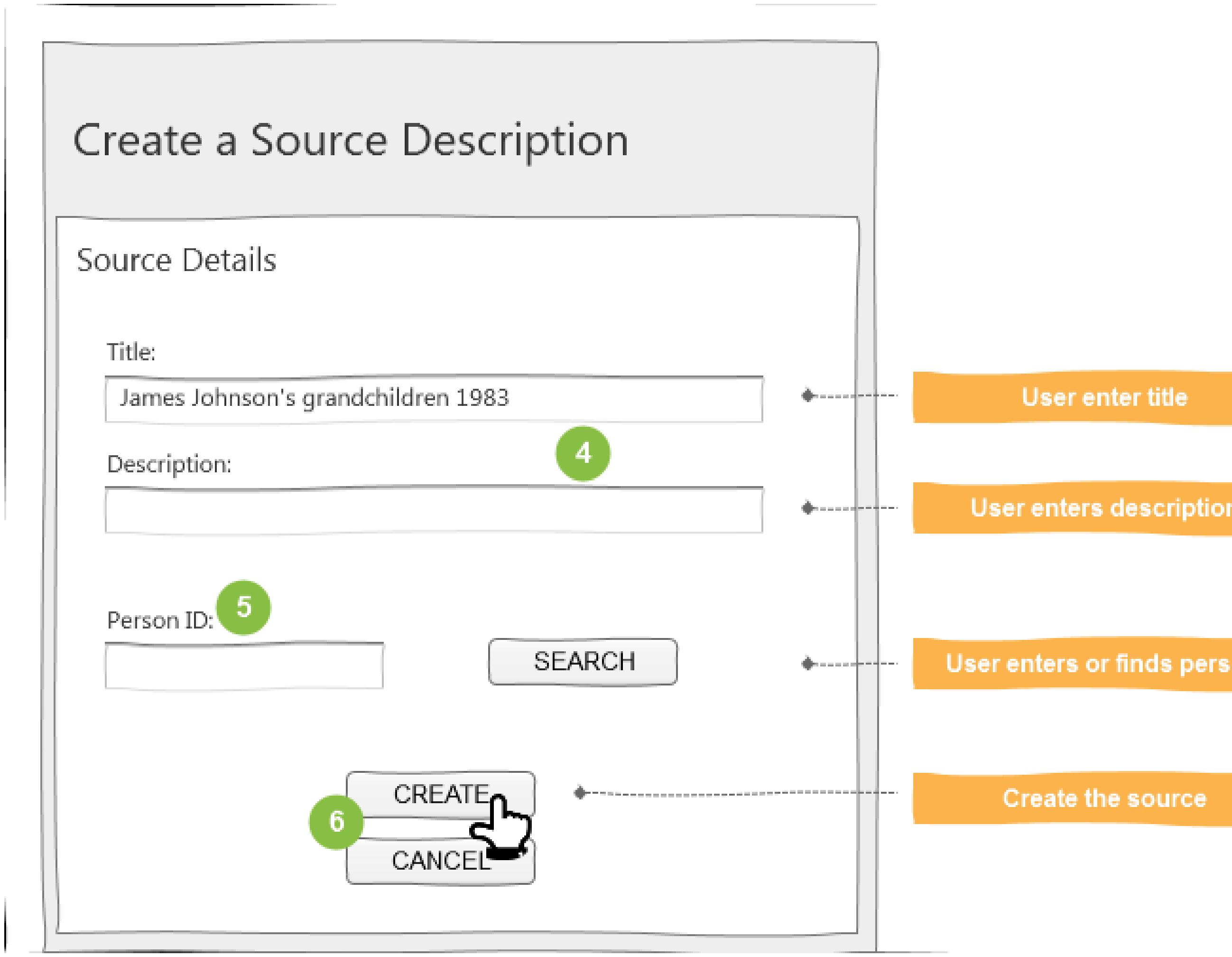
1. Display a list of memories.
2. The user selects a memory from the list.

3. The user chooses to create a source.



4. The user gives the source a title and description.
5. The user provides or finds a person ID.

6. Allow the user to create (or cancel) the source.



The memory is created as a source.

See Also

- Read memories list
- Read memory

Delete a Memory Starting from a Memory List

Users can delete memories, starting from a list of the memories that the user created.

Note: When you delete a memory, it is removed entirely from the database. When you detach a memory you simply remove the reference that connects the memory to a person in Family Tree. A detached memory is still in the database and can be connected to other persons.

Assumptions

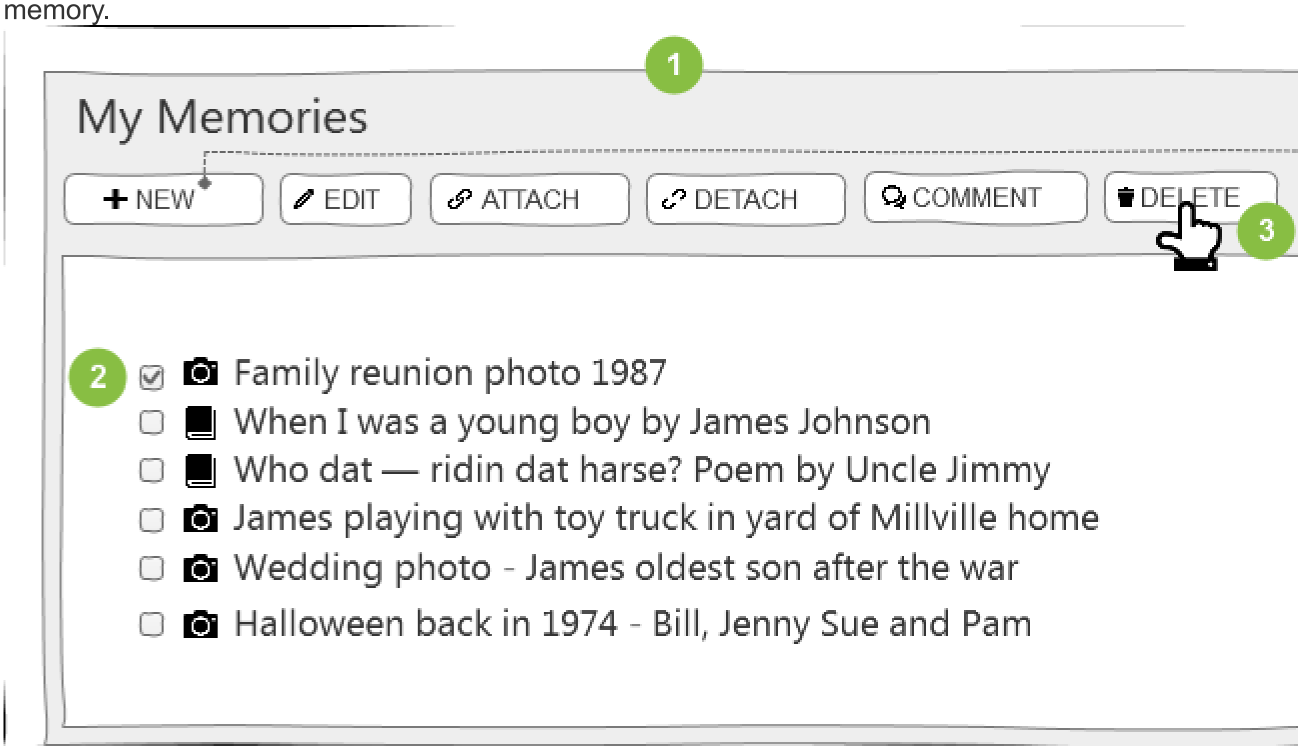
- The user is signed in to FamilySearch.org through the third-party application.

Programming Steps

1. Display memories that the current user created.
2. Allow the user to select a memory to delete.
3. Delete the memory that the user selected.

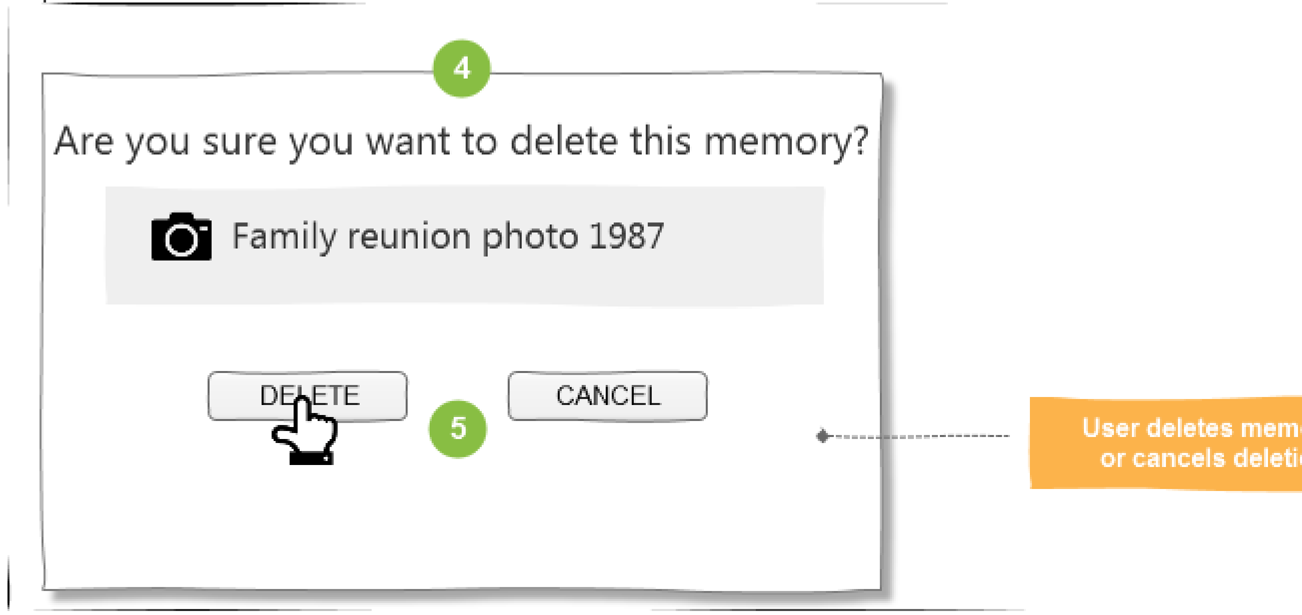
User Interface Suggestion

1. Display the memories that the user has created.
2. The user selects a memory to delete.
3. The user selects an option to delete the memory.



4. Ask the user to confirm or cancel the deletion request.

5. The user confirms or cancels the deletion request.



6. Return the user to the list of memories. The deleted memory is no longer shown in the list.

See Also

Read memories

Delete Personas to Detach Memories from Persons in Family Tree

When you delete a persona, you effectively detach a tree person from a memory.

Note: When you detach a memory, you simply remove the reference that connects the memory to a person in Family Tree. A detached memory is still in the database and can be connected to other persons. When you delete a memory, it is removed entirely from the database.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- The user has a person ID or is allowed to find one.

Programming Steps

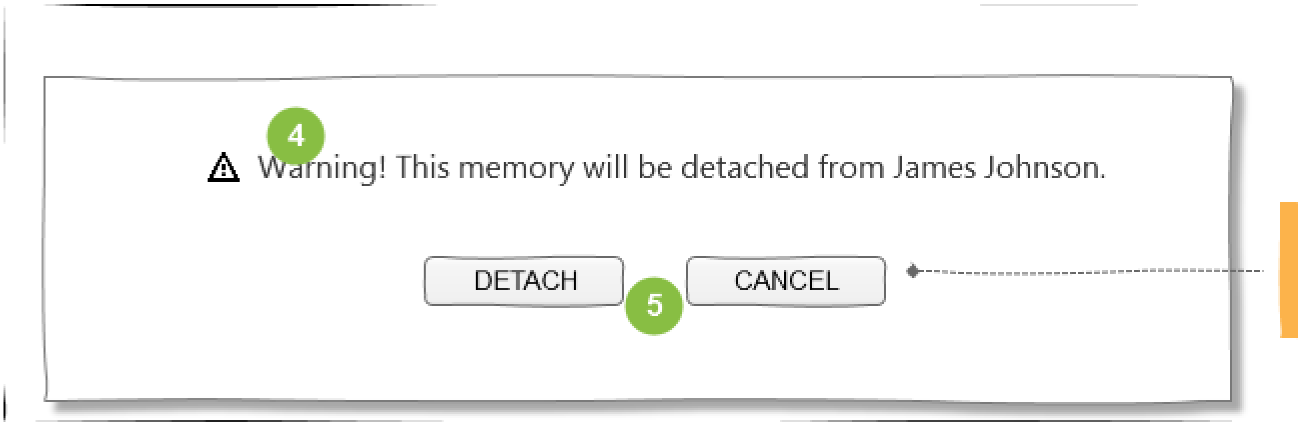
1. Read the memories that are attached to a specific person and display them.
2. Detach the memory that the user selects.

User Interface Suggestions

1. Display the memories that are attached to a person in Family Tree.
2. The user selects a memory.

3. The user selects an option to detach the memory.

4. Display a verification warning message.
5. The user detaches the memory or cancels the request and is returned to the list of memories.



See Also

Read memories

Delete a Memory Comment Starting from a Memory List

Users can delete memory comments that they created.

Assumptions

- The user is signed in to FamilySearch.org through the third-party application.
- The app has already listed the memory comments that the authenticated user contributed.

Programming Steps

1. Allow the user to select a memory comment from the list of comments the user created.
2. Delete the memory comment that the user selected.

User Interface Suggestion

1. Display the comments contributed by the current user.
2. The user selects a memory comment to delete.
3. The user selects an option to delete the memory.
4. Ask the user to confirm or cancel the deletion request.
5. The user confirms or cancels the deletion request.
6. Return the user to the list of memory comments. The deleted memory comment is no longer shown in the list.

Audio Recordings Guidelines

FamilySearch audio recordings should allow individuals to share audio recordings about ancestors that help the family come to know and love one another. The following guidelines should apply:

Uploading

1. **Appropriate.** Content should support appropriate standards of modesty and virtue.
2. **Relevant.** Content should support a family history purpose.
3. **Heart-turning.** Content should support individuals coming to know and love their ancestors.
4. **Noncommercial.** Content should not advertise or promote products or infringe on intellectual property rights.

Audio Recordings Screening

Audio recordings MAY NOT include any of the following content:

1. Sacred or other religious rites that desecrate an associated religious belief.
2. Immoral acts or immodest behavior.
3. Obscene or profane words of any kind.
4. Product or website endorsements.
5. Illegal acts, products, or services. This includes drugs.
6. Vulgarity or vulgar language.
7. Expressions of hatred or discrimination.
8. The promotion of racism.
9. Violence or mutilation.
10. The promotion of current productions, causes, or charities.
11. Commercial audio recordings. The only exceptions are if:
    - You have legal rights to the content.
    - The copyright has expired.

Guidelines for Documents

FamilySearch Documents should allow individuals to share documents and stories of ancestors that help the family come to know and love one another. The following guidelines apply to the content of uploaded documents:

Uploading

1. **Appropriate.** Content should support appropriate standards of modesty and virtue.
2. **Relevant.** Content should support a family history purpose.
3. **Heart-turning.** Content should support individuals coming to know and love their ancestors.
4. **Noncommercial.** Content should not advertise or promote products or infringe on intellectual property rights.

Documents Screening

Documents MAY NOT include any of the following images, depictions, or descriptions:

1. Nudity or immodesty. Clothing depicted in photos should be zipped and buttoned properly.
2. Sacred or other religious clothing or rites that desecrate an associated religious belief.
3. Cross-dressing.
4. Links to outside websites (no URLs).
5. Obscene or profane words of any kind.
6. Body shots without a head or face.

7. Product or website endorsements. Logos and wording on clothing or on billboards, trucks, etc. will be evaluated individually.
8. Illegal acts, products, or services. This includes drugs.
9. Vulgarity or vulgar gestures.
10. Expressions of hatred or discrimination.
11. Violence and mutilation.
    - War, hunting, and fishing photos will be evaluated individually.
    - Photos with piercings and tattoos will be evaluated individually.
12. The promotion of racism.
13. The promotion of current productions, causes, or charities.
14. Commercial photos. The only exceptions are if:
    - You have legal rights to the photo.
    - The copyright has expired.

Photos in documents may not be edited in such a way as to make them inaccurate, false, or misleading. Also, any group shot photo in a document that contains even one individual that violates any of the guidelines in this document is not allowed.

Guidelines for Photos

FamilySearch Photos should allow individuals to share photos of ancestors that help the family come to know and love one another. The following guidelines should apply to photos:

Uploading

1. **Appropriate.** Support appropriate standards of modesty and virtue.
2. **Relevant.** Support a family history purpose.
3. **Heart-Turning.** Support individuals coming to know and love their ancestors.
4. **Noncommercial.** Do not advertise or promote products. Do not infringe on intellectual property rights.

Photo Screening

1. No nudity or immodesty. Clothing should be zipped and buttoned properly.
2. No sacred or other religious clothing is allowed if it is displayed in a way that desecrates an associated religious belief.
3. No cross-dressing.
4. No body shots without a head or face.
5. No product or website endorsements. Logos and wording on clothing or on billboards, trucks, etc. will be evaluated individually.
6. No profanity.
7. No images of illegal acts, products, or services. This includes drugs.
8. No vulgarity or vulgar gestures.

9. No expressions of hatred or discrimination.
10. No images of violence and mutilation.
    - War, hunting, and fishing photos will be evaluated individually.
    - Photos with piercings and tattoos will be evaluated individually.
11. No promotions of racism.
12. No promotions of current productions, causes, or charities.
13. No commercial photos. The only exceptions are:
    - You have legal rights to the photo.
    - The copyright has expired.
14. Photos edited in such a way as to make them inaccurate, false, or misleading are not allowed.

Any group shot that contains even one individual that violates any of the guidelines in this document is not allowed.

Write Sources Implementation Guide

Create a Source

Users can programmatically create sources that are placed in a collection in their own source box using the source resource of the api or the application can post to Create Source Destination Page.

Assumptions

- The user is signed in and authenticated with FamilySearch through the third-party application.

Programming Steps

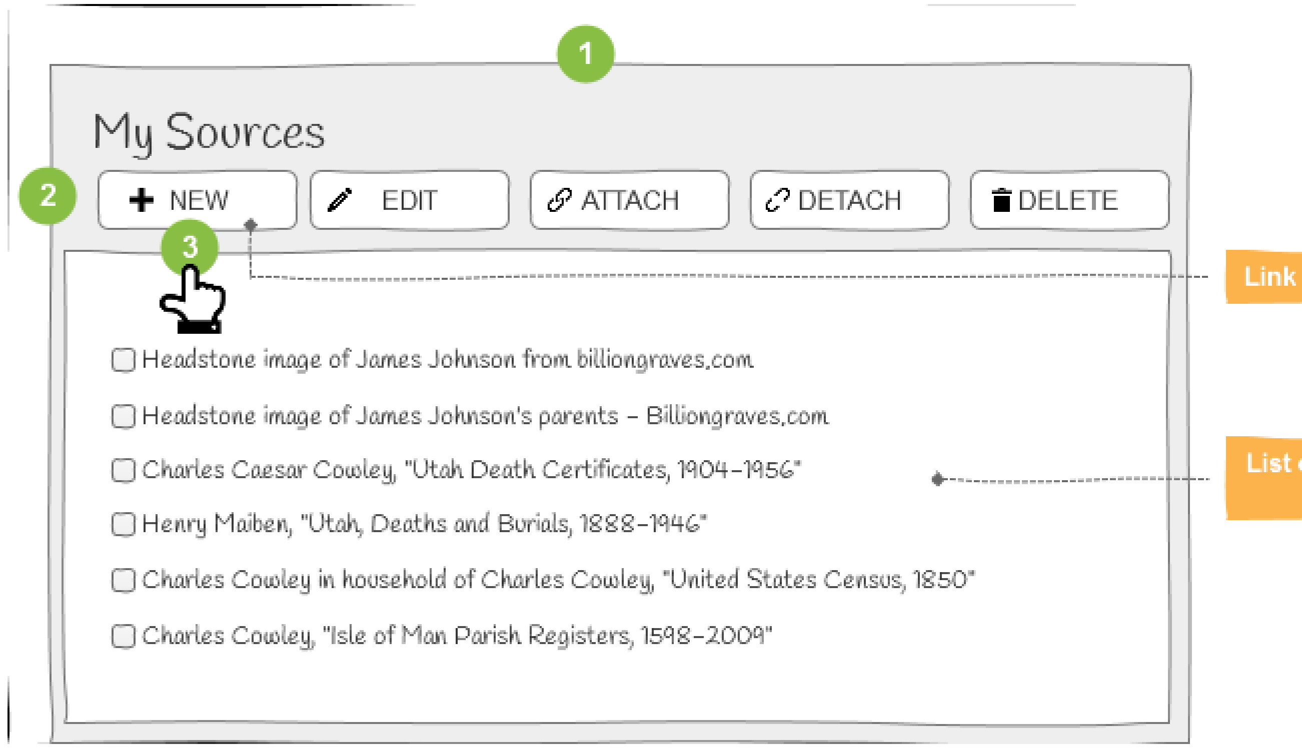
1. List Sources created by the current user.
2. Collect source description information.
3. Create the source.
4. Save the srcID.

User Interface Suggestions

This example shows how to initiate creation of a source while viewing a list of sources. You could initiate creation of a source while viewing the details of a person. There are many places in your application you could initiate creation of a source.

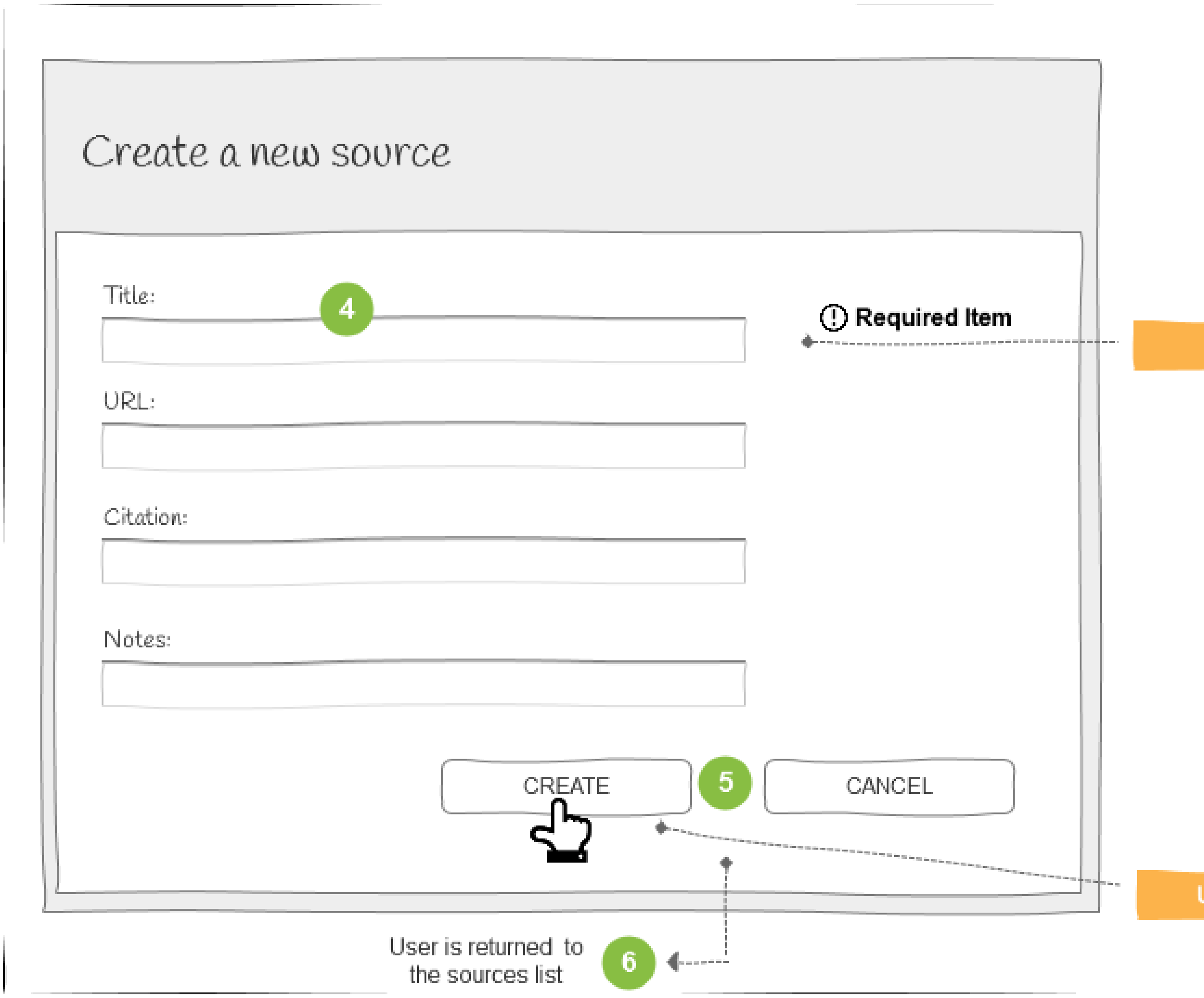
1. Display the list of current sources in the user's source box.
2. Show the options to create, edit, attach, detach, and delete sources.

3. The user selects the option to create a new source.



4. Prompt the user to enter the source title, URL, citation, and notes.
   Note: The title is the only required field.
5. Save the source or cancel.

6. Return to the list of sources.



Attach a Source to a Person

A source can be attached to a person in Family Tree. Attaching a source is accomplished by creating a source reference.

Assumptions

- The user is signed in to FamilySearch through the third-party application.

Programming Steps

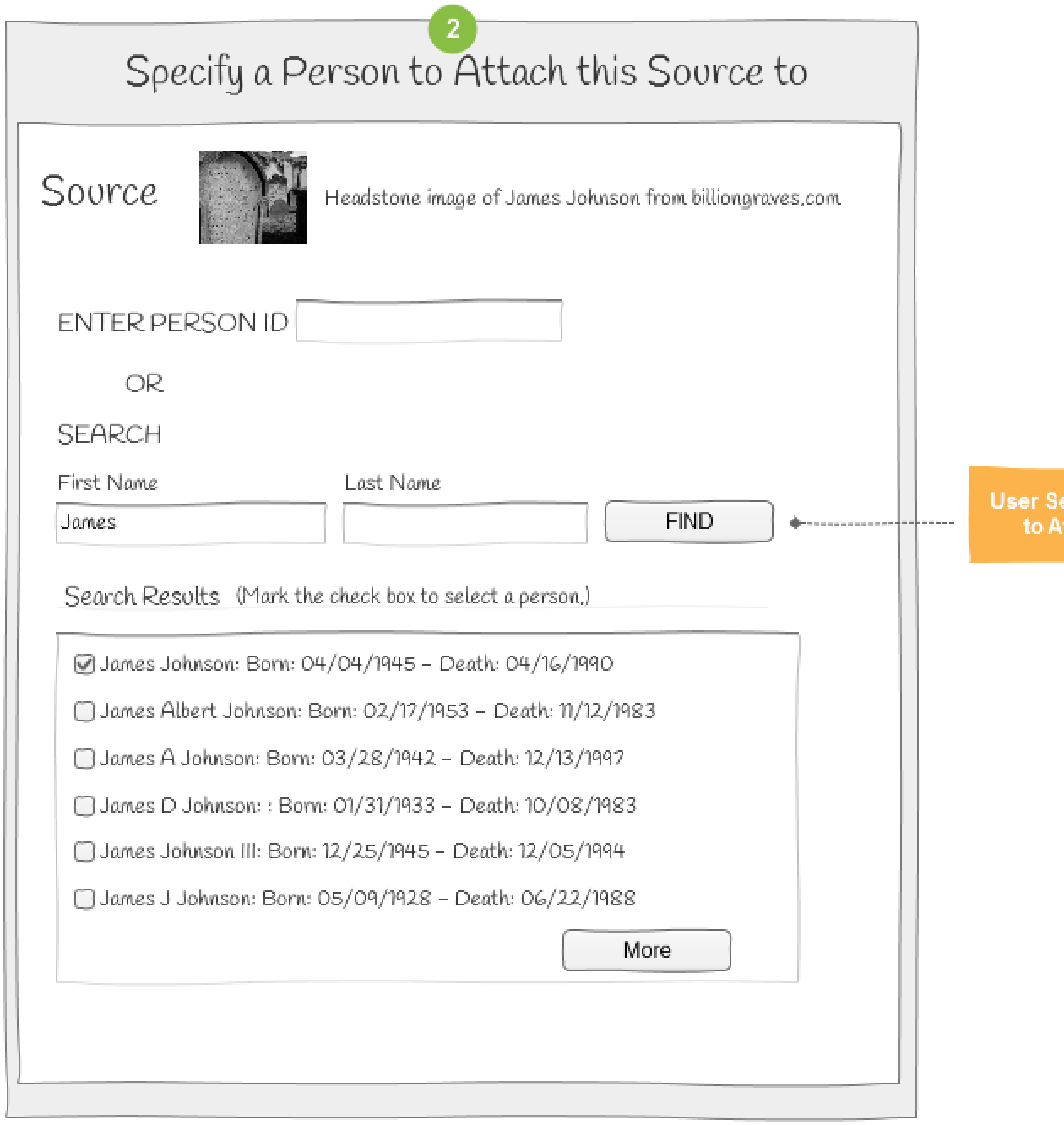
1. Select a source to attach.
2. Select a person to attach the source to, and specify which vitals to tag with the source.

3. Prompt the user to enter an explanatory note.
4. Attach the source to the person.

User Interface Suggestions

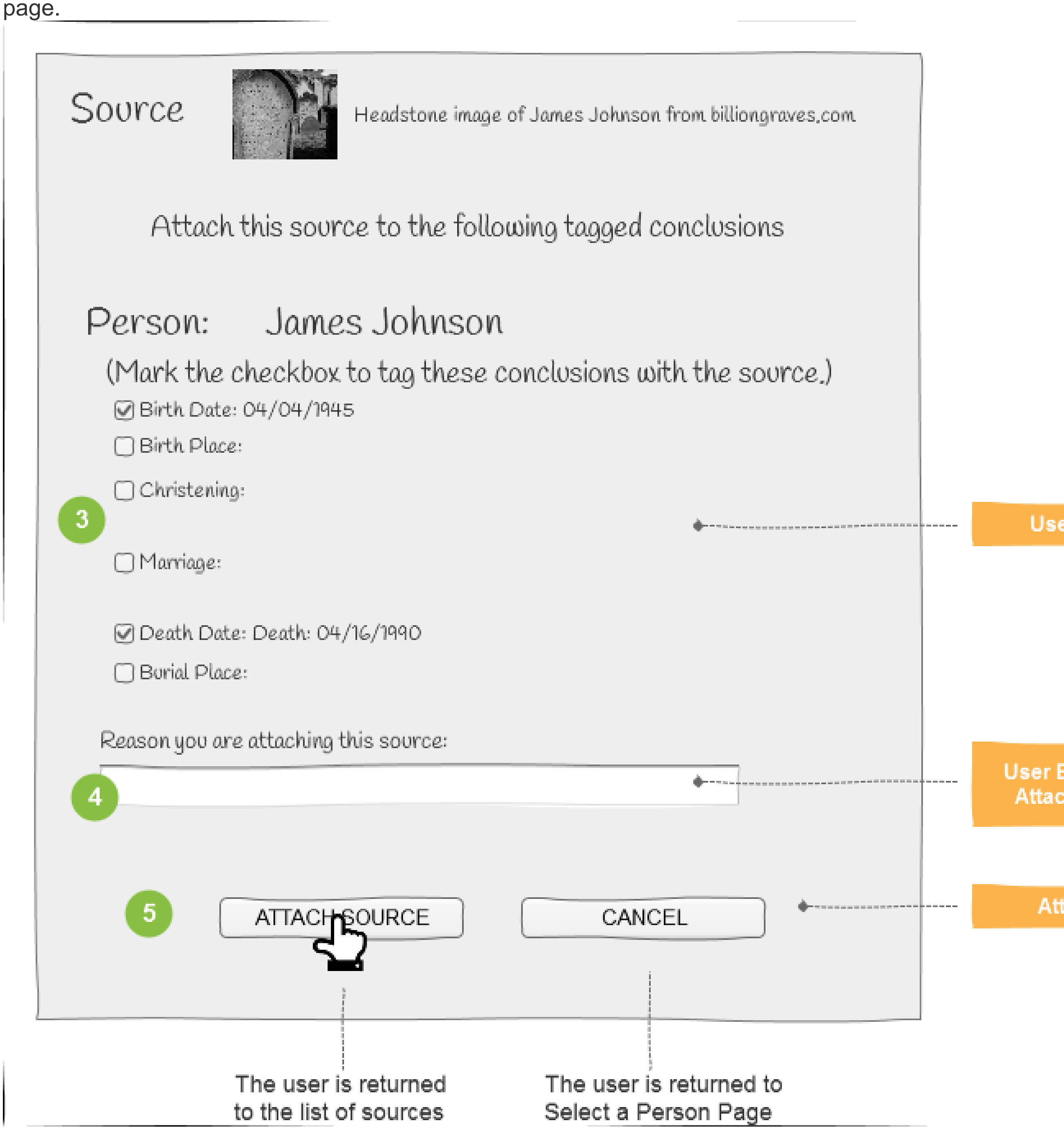
1. Select a source from a list of sources, and click Attach.

2. The user searches for and selects a person to attach to or enters a Person ID.



3. The user specifies the conclusions to tag with this source.
4. Prompt the user to enter a reason for attaching the source.
5. Attach the source and return to the sources list, or cancel and return to the Person selection

page.



Edit an Existing Source Starting from a Source List

Sources can be edited.

Assumptions

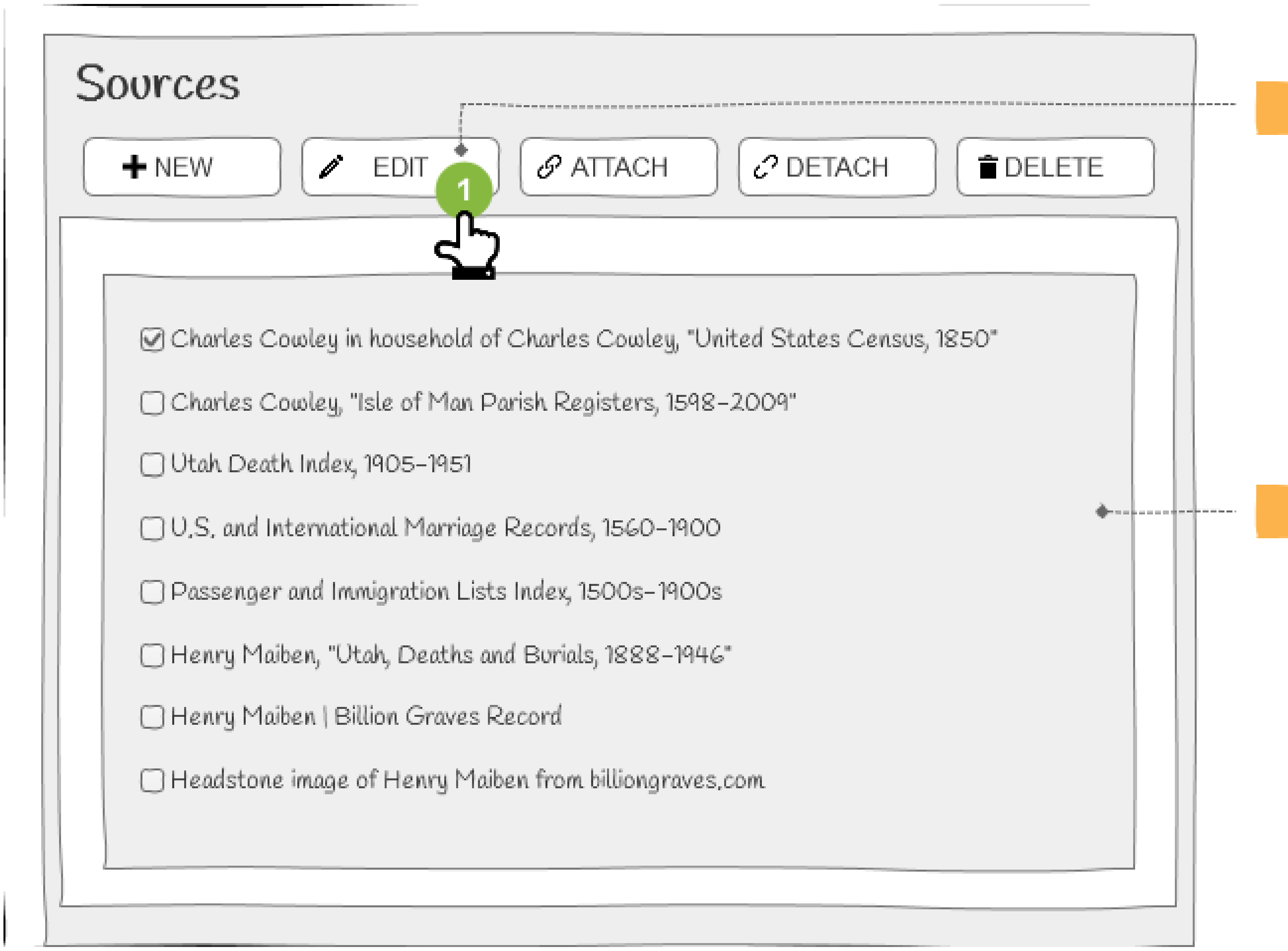
- The user is signed in to FamilySearch through the third-party application.

Programming Steps

1. List Sources, and select a source to edit.
2. Display the source information, and allow editing.
3. Save edits.

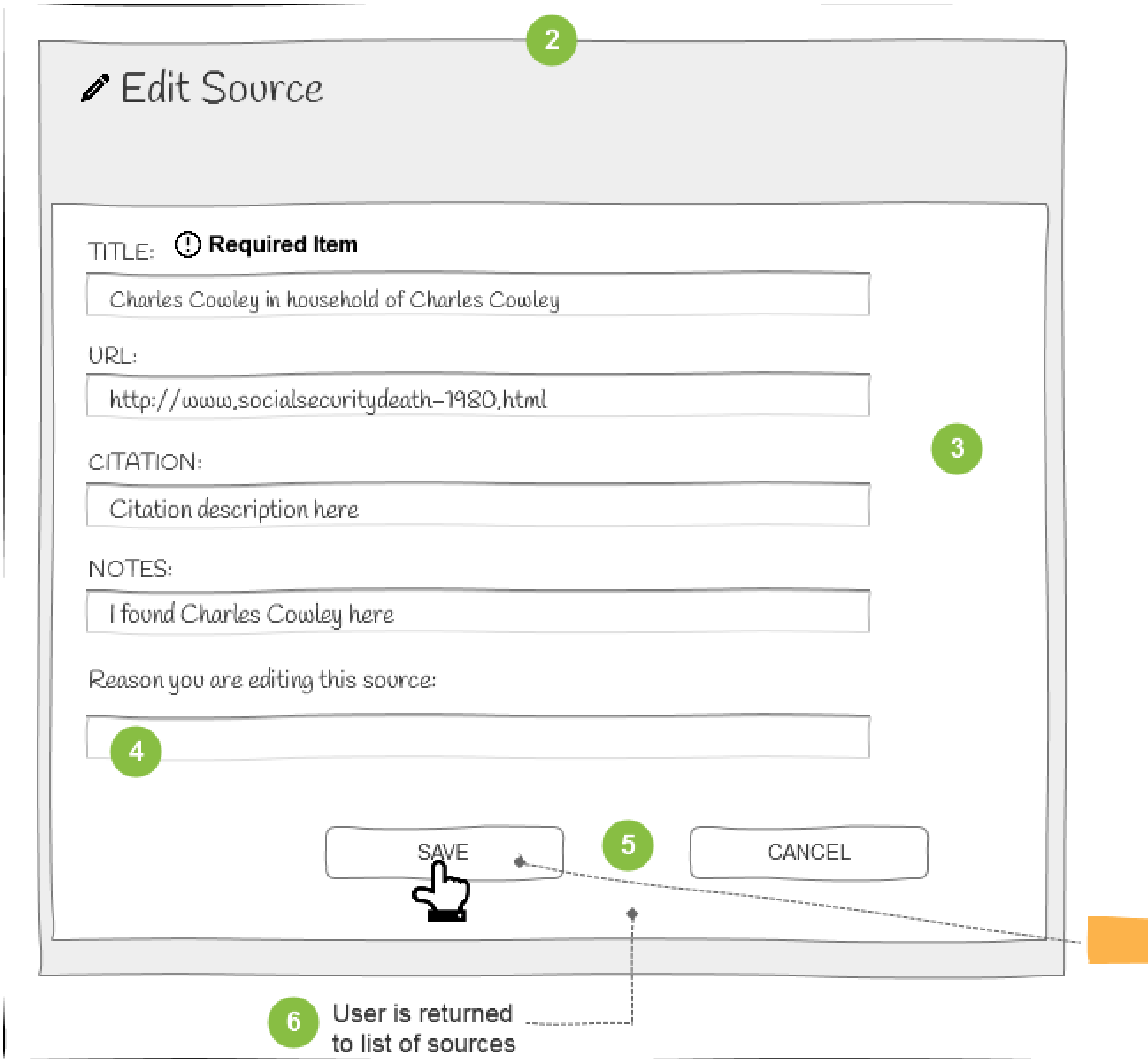
User Interface Suggestions

1. Select a source from a list of sources, and click Edit.



2. Display the selected source title, URL, citation, and notes. (Optional) Display the following information if available:
    - Contributor's name
    - Contribution date
    - Reason statement
    - List of persons and relationships that the source is attached to.
3. Allow the user to modify the source title, URL, citation, and notes.
   Note: The title is the only required information.

4. Prompt the user to enter a reason for the changes.
    - Show the prior reason, if available, in the reason entry box
    - The label of this entry box should be “Why this edit is correct”
    - The user can add to it, write over it or delete it.
5. Save or cancel the changes.
6. Return to the list of sources.



Detach an Existing Source

Sources can be detached from persons. Detaching a source from a person is to delete a source reference. That is, when you delete a source reference, you are detaching the source from the person.

Assumptions

- The user is signed in to FamilySearch through the third-party application.

Programming Steps

1. Read the list of sources the user has created. Display the sources, and prompt the user to select one to detach.
2. Read all persons who are attached to the targeted source. Display the persons, and prompt the user to select one to detach from.
3. Detach the selected source from the selected person.

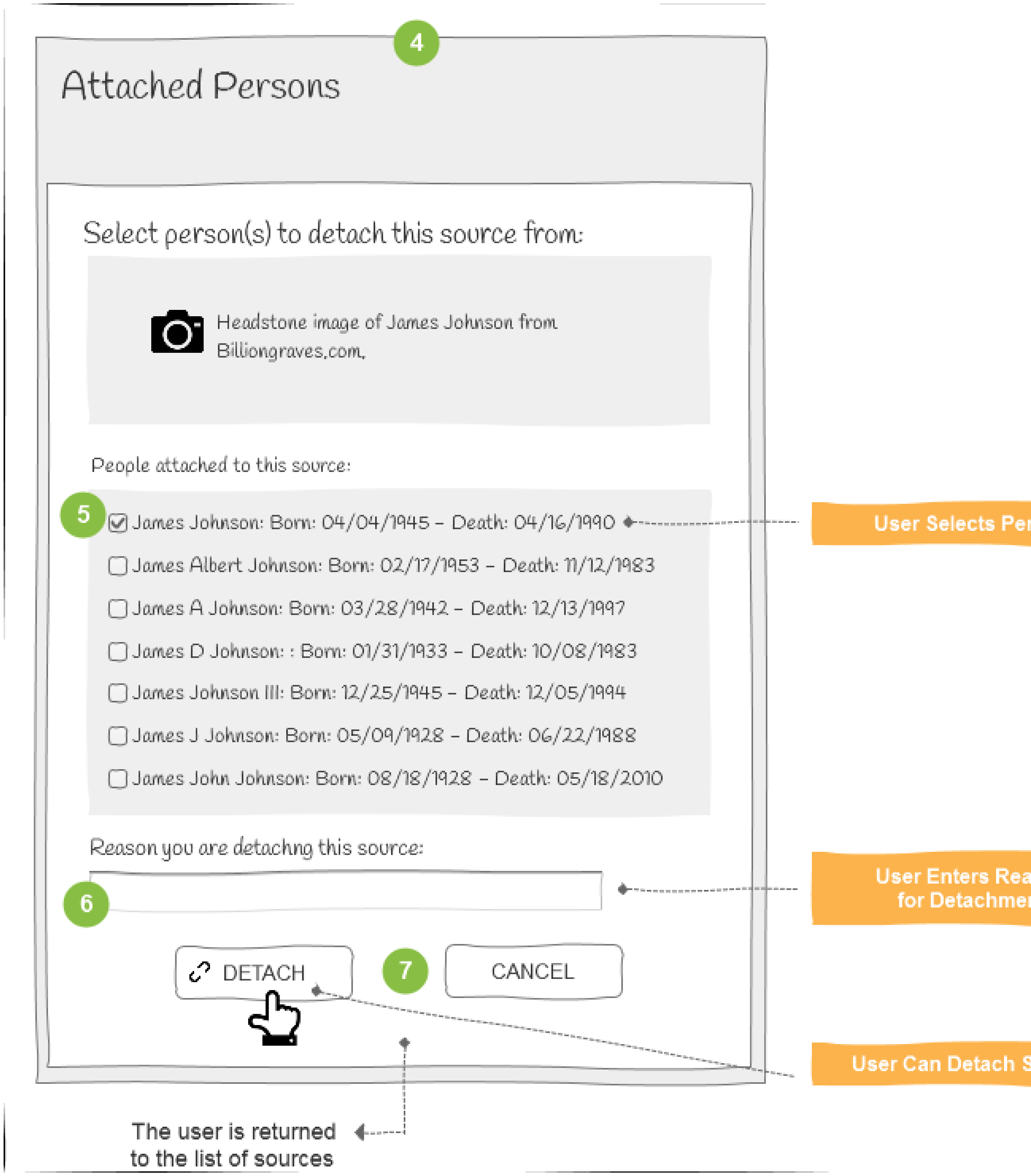
User Interface Suggestions

1. Display the list of sources that a user has created.
2. The user selects the source to be detached.

3. The user clicks Detach.

4. Display the persons who are attached to the selected source
5. The user marks the person to detach from.
6. Prompt the user to enter a reason why the source is being detached.
7. The user clicks DETACH or CANCEL.
   Detach the source, or cancel and return to the list of

sources.



Delete an Existing Source

Sources can be deleted by the user who created them.

Assumptions

- The user is signed in to FamilySearch through the third-party application.

Programming Steps

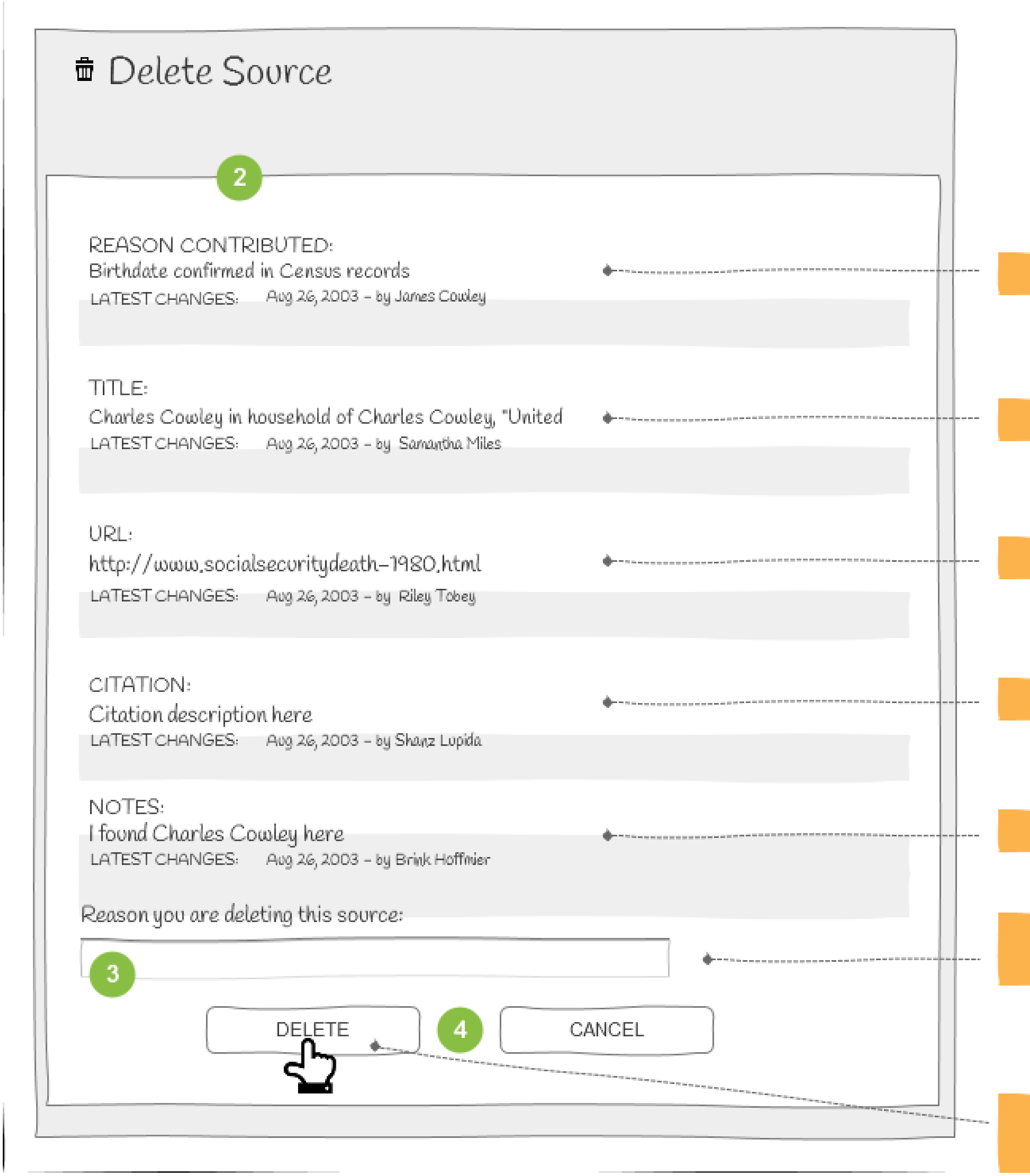
1. Read the sources created by the current user.
2. Read a selected source, and display the details, including all persons and relationships the source is attached to.
3. Delete the source if you are sure it is not relevant to any and all the referenced persons and relationships.

User Interface Suggestions

1. Select a source from a list of sources, and click Delete.

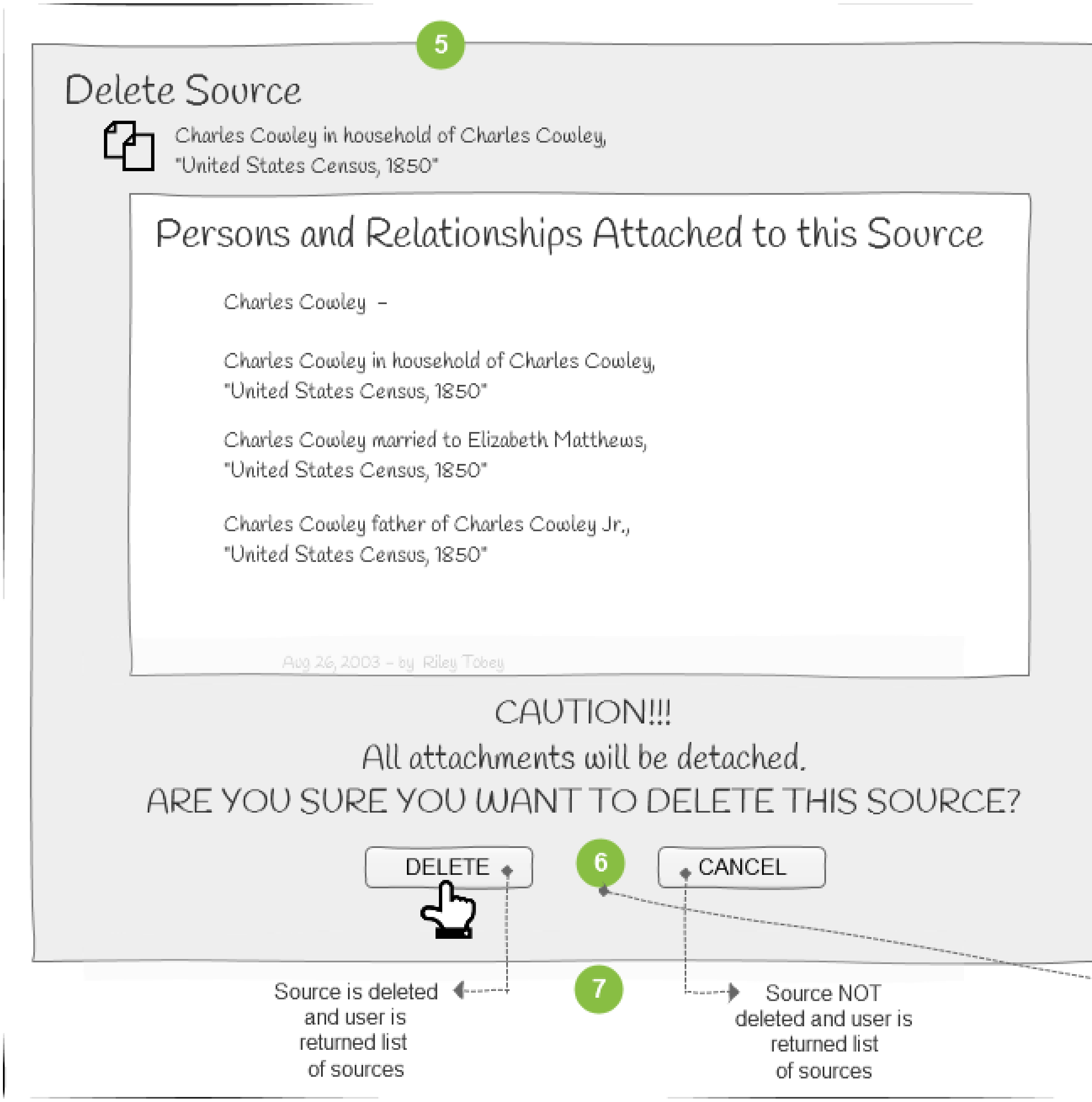
2. Display the source details.
3. Prompt the user to enter a reason why the source is being deleted.

4. The user clicks DELETE or CANCEL.



5. Display a confirmation page with a list of all persons and relationships attached to the source.
6. Require confirmation to delete the source, or cancel.

7. Delete the source, or cancel and return to the list of sources.



Side-by-Side Comparison of Sources from Separate Trees

Source information found in two separate databases can be displayed side-by-side for easy comparison. Each side can show source details, and the differences can be visually enhanced. Users can be enabled to add, replace, delete, or detach source data from one database to the other. When updating the FamilySearch database, the contributing source detail must be viewable with the contribution date and the contributor's name. The reason for the change should be entered by the user before finalizing the change. Try to map with as much fidelity as possible from citation to Source fields so the user avoids any duplication of sources and keeps the FamilySearch sources in "sync" with the partner's product. Most important are the Title, URL followed by and Citation and Note.

Assumptions

- The user is logged in and authenticated to both FamilySearch and the third-party application.

Programming Steps

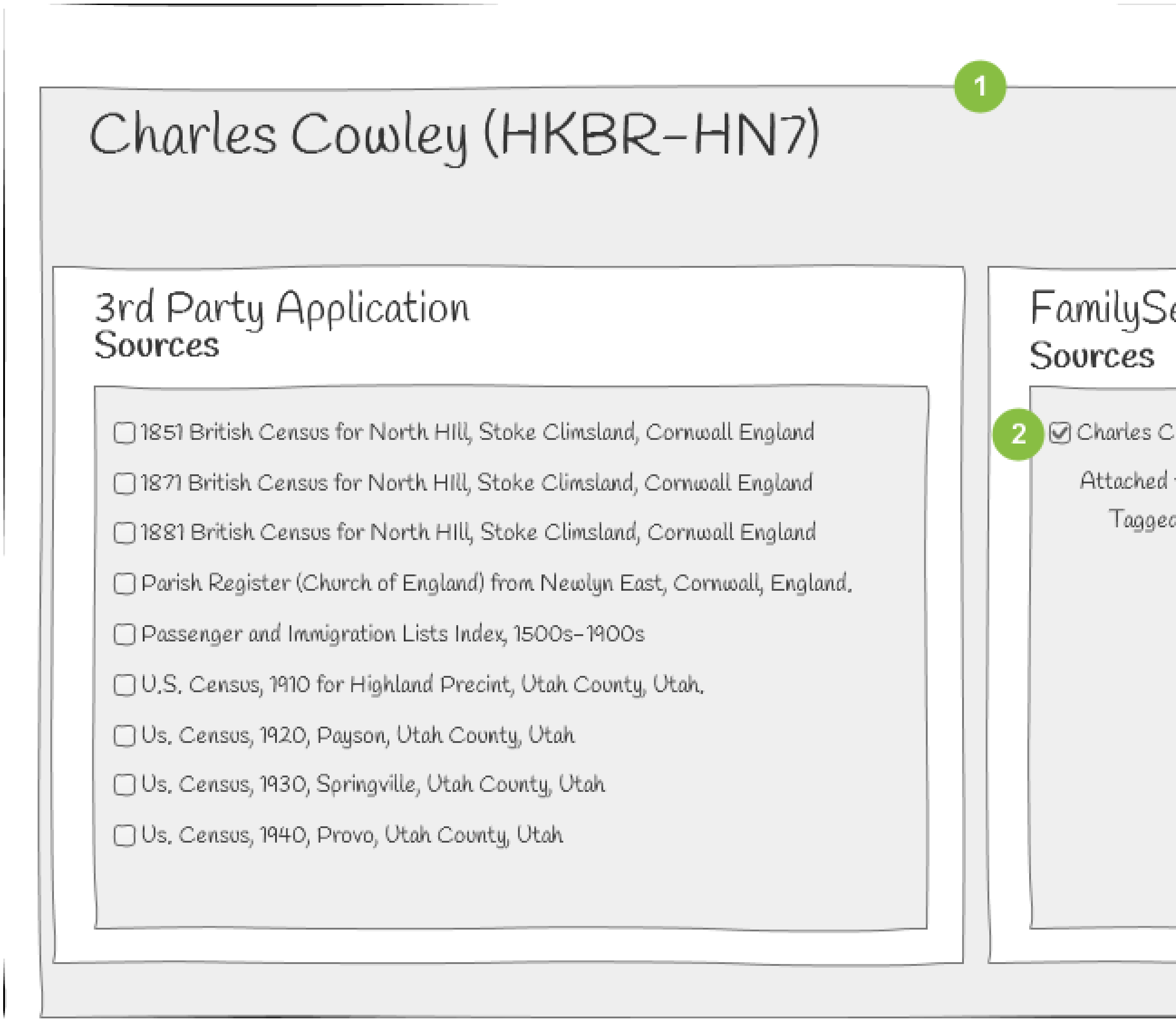
1. Read and display the sources from each database side-by-side. Highlight differences.
2. Collect instructions from the user.
3. Summarize and ask for user confirmation or to cancel.
4. Update the data using match SrcIDs. Use FamilySearch APIs to update FamilySearch data.
5. Read and display the sources from each database side by side again.

User Interface Suggestions

Update Partner Data from FamilySearch Data

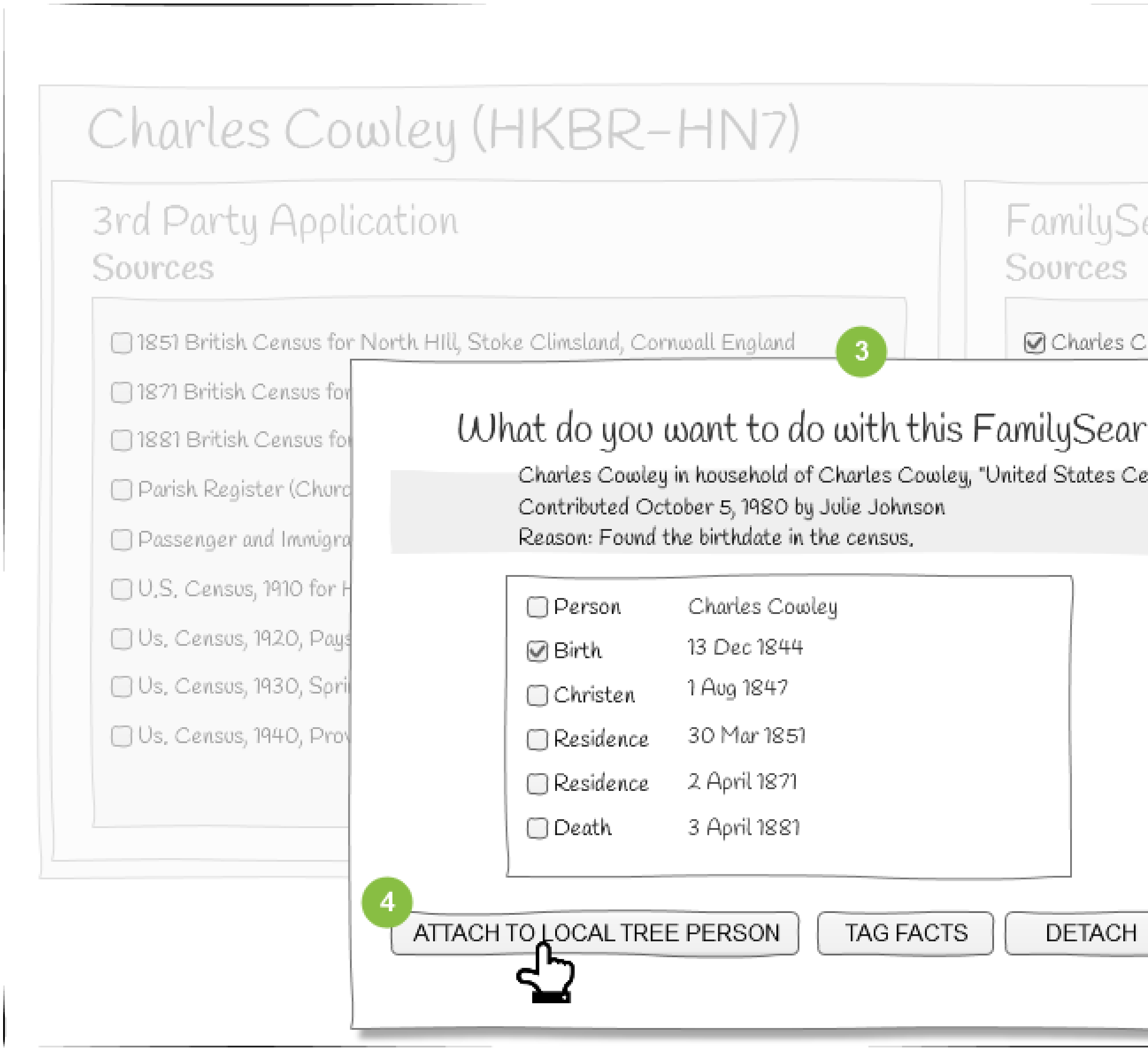
1. Display the data from each database side by side or have an icon or button to display the details.

2. Using a checkbox or icon, the user selects a FamilySearch source to copy to the local database.

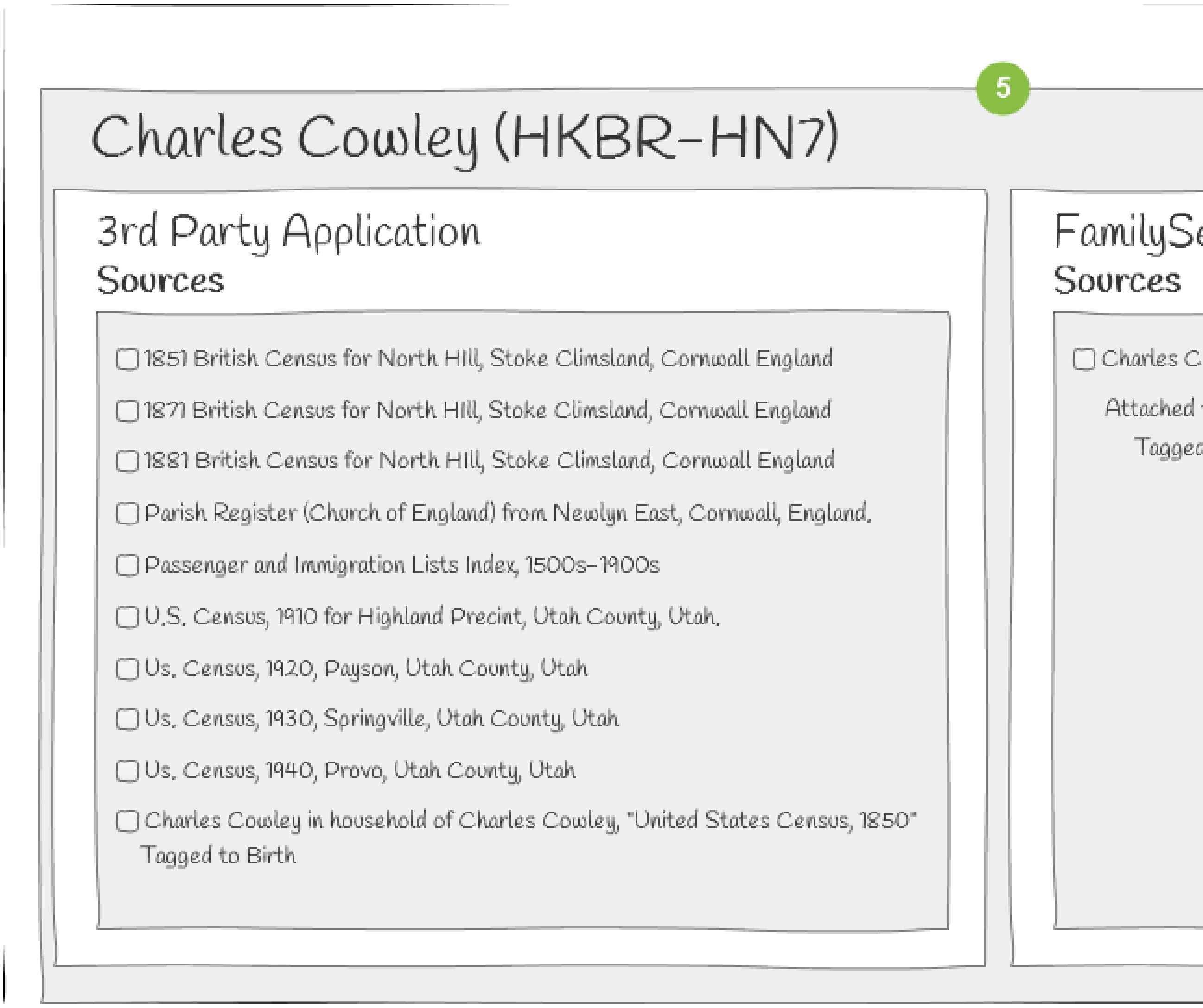


3. Display the prior values, contributor, date, and reason.

4. The user designates the action to take.



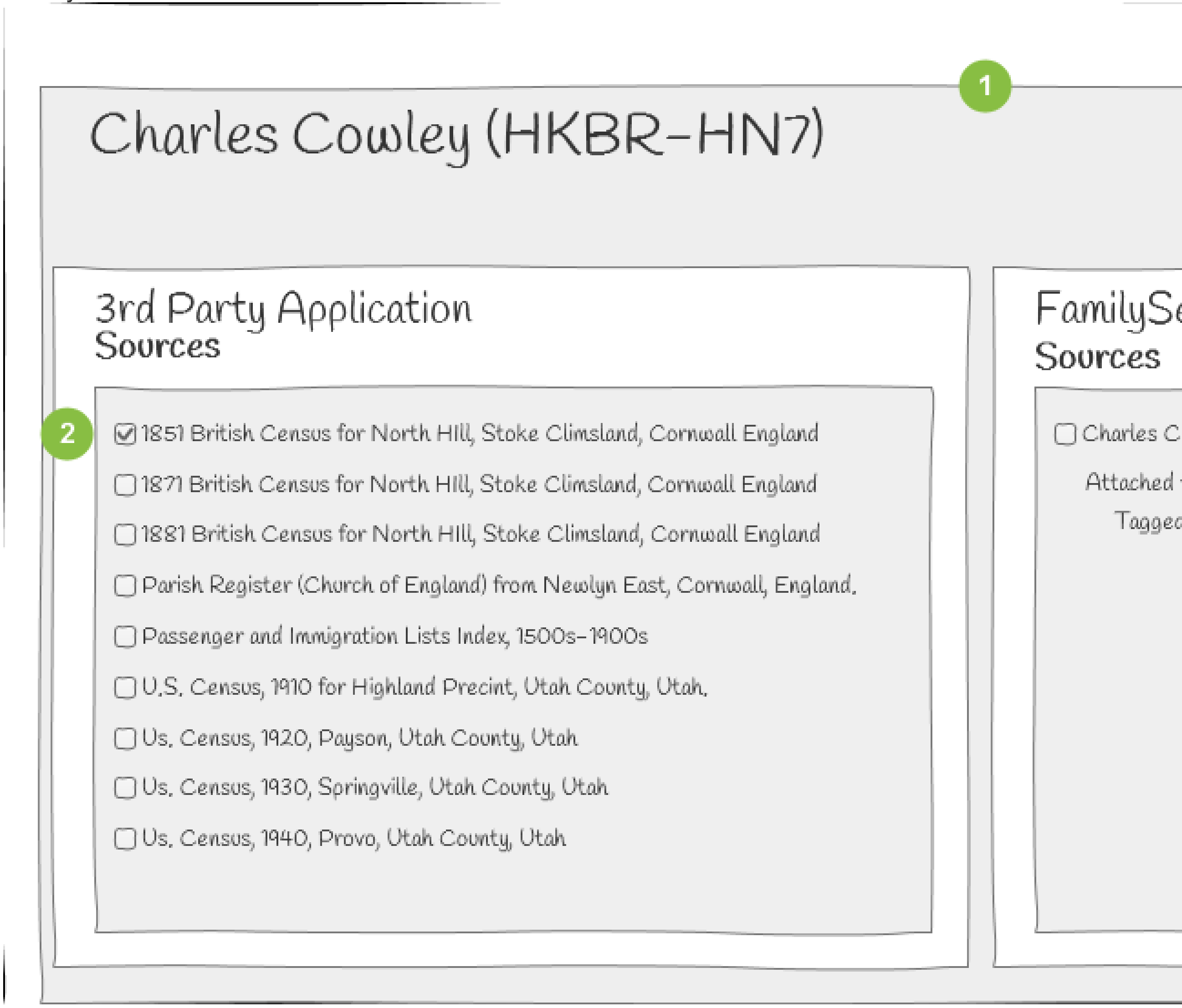
5. Display the changes on the partner side.



Update FamilySearch Data from Partner Data

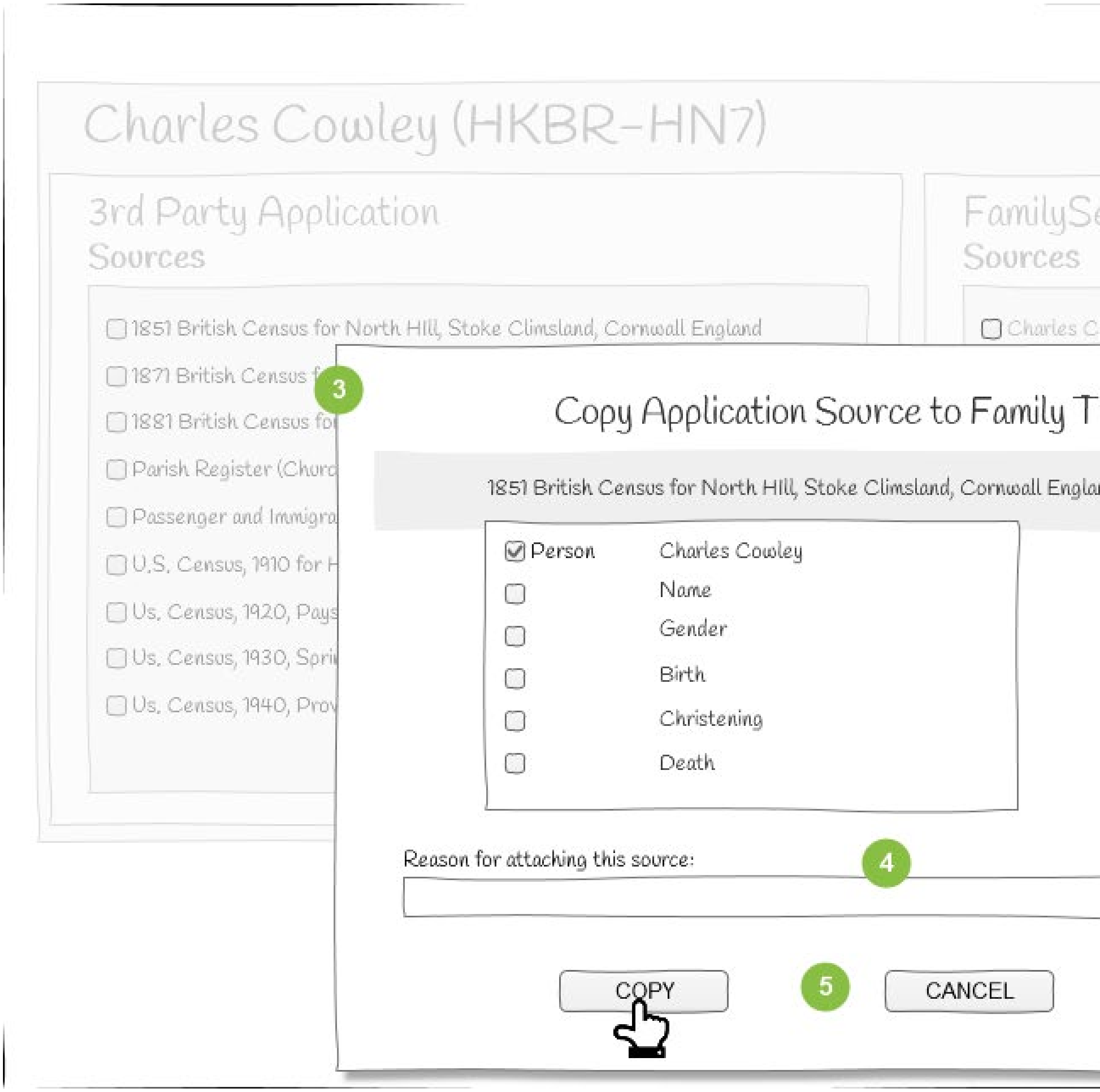
1. Display the data from each source side by side or have an icon or button to display the details.

2. Using a checkbox or icon, the user selects a local source to copy to FamilySearch.

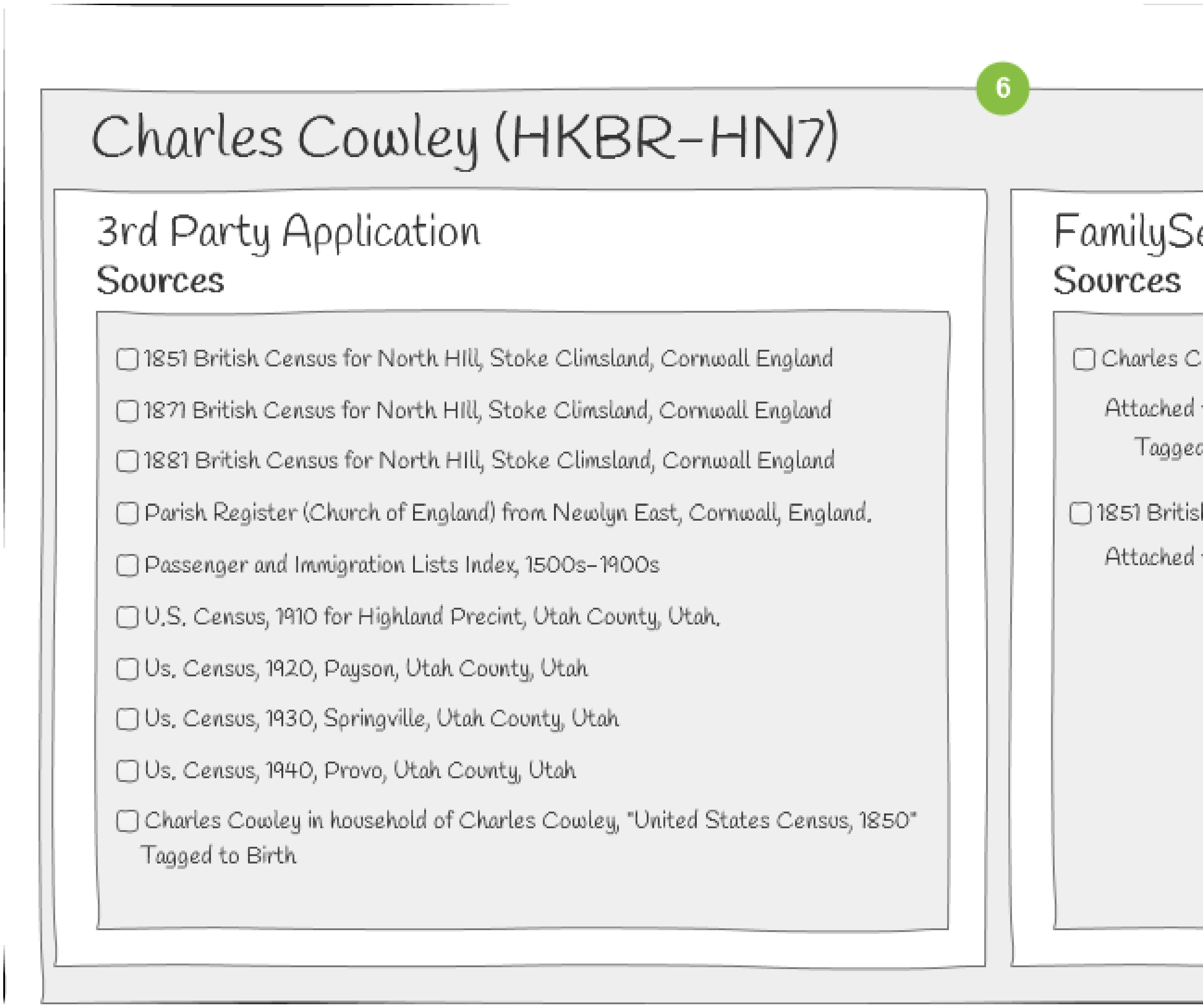


3. Display the application source values.
4. Prompt the user to enter a reason for the changes.
   - Show the prior reason, if available, in the reason entry box
   - The label of this entry box should be “Why this edit is correct”
   - The user can add to it, write over it or delete it.

5. The user confirms the copy or cancels.



6. Display the changes on the FamilySearch side.



Write Trees Implementation Guide

Modify a Person in FamilySearch

An existing person's information in FamilySearch can be changed.

Assumptions

- The user must be signed in to FamilySearch through your application.
- The user must select the option to edit the person information.

Programming Steps

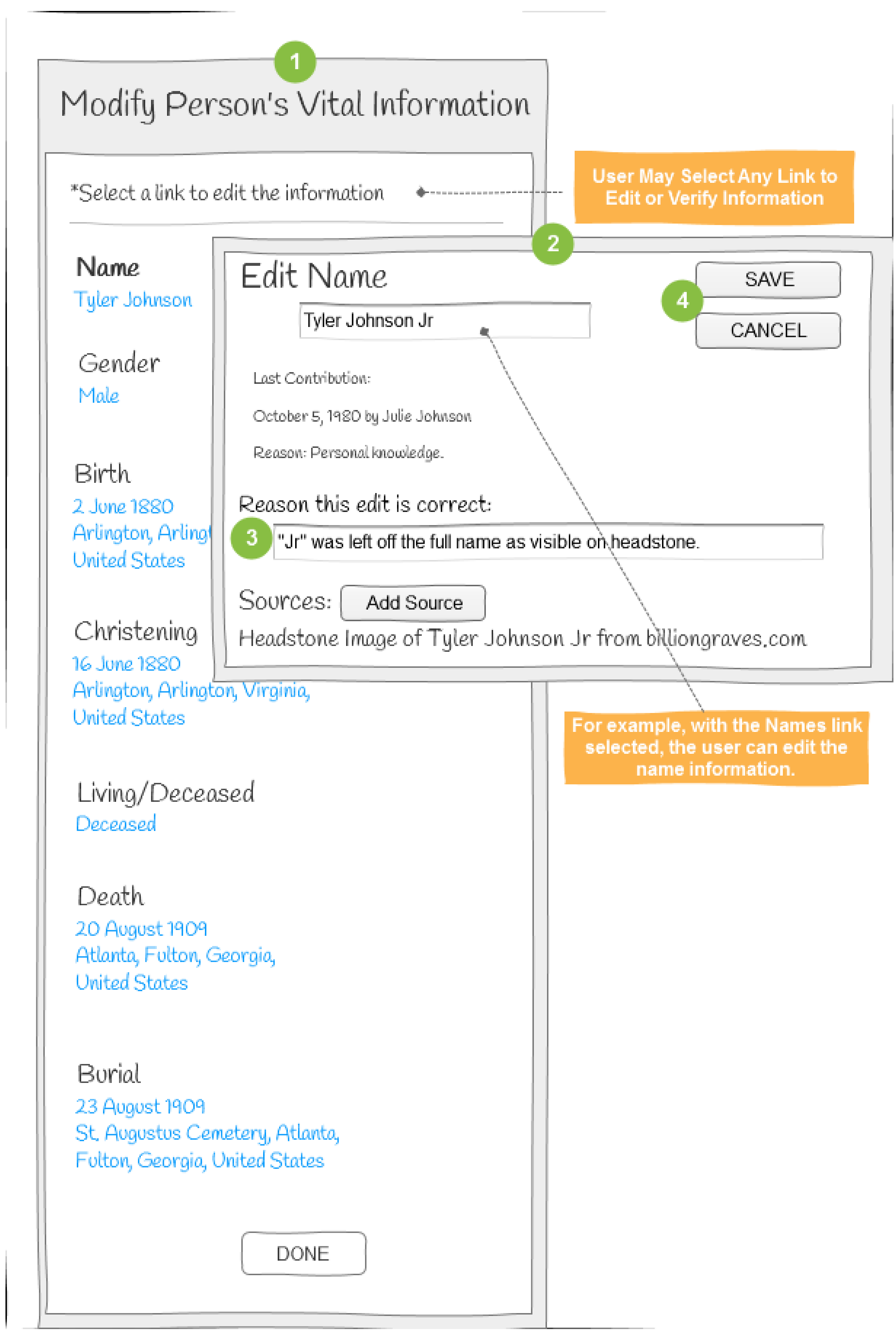
1. Read the existing values for the person.
2. Give the user the option to edit current values.
3. Post the updated information.

User Interface Suggestions

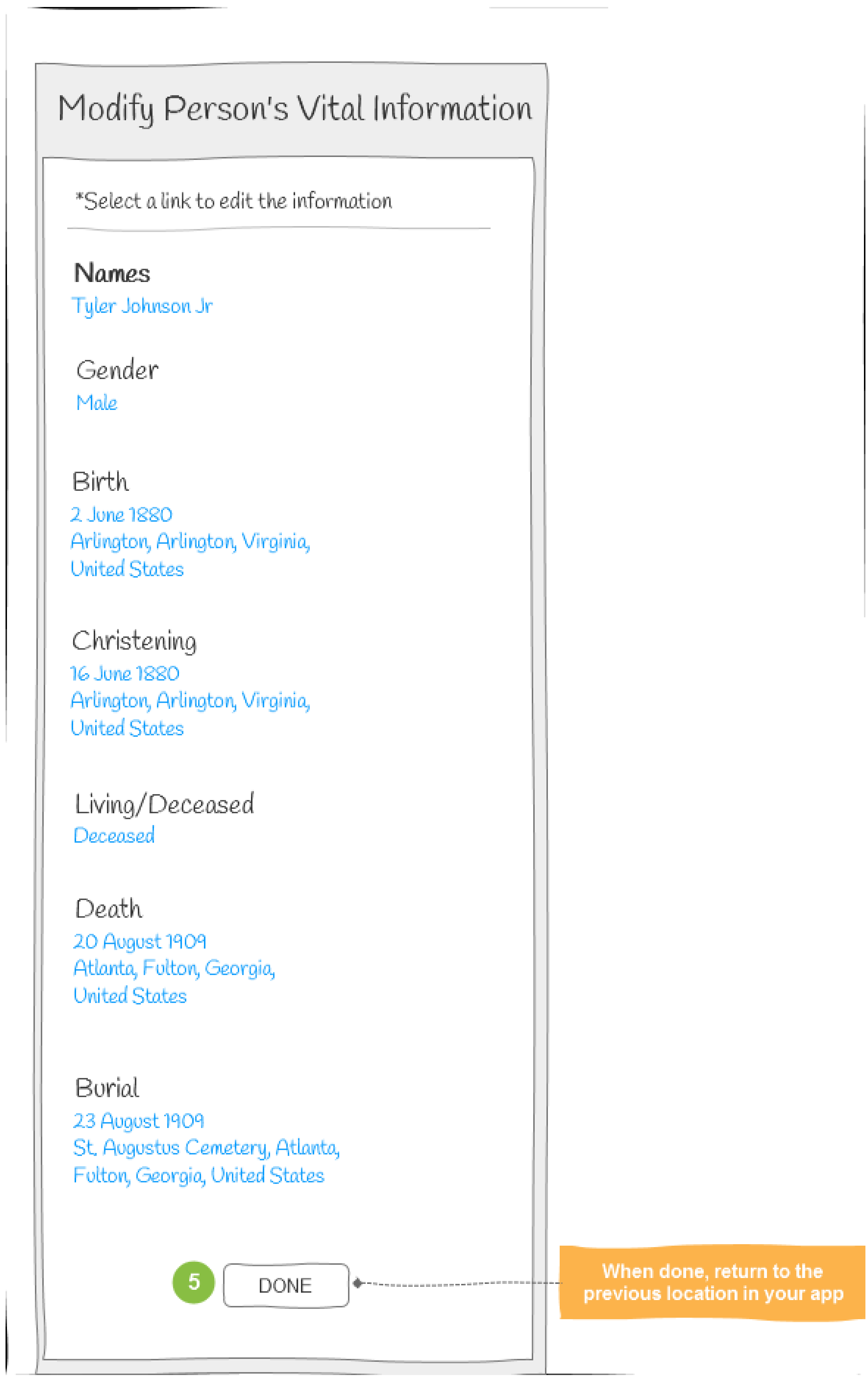
To modify vital information of a person, comply with the following user interface guidelines.

1. Prompt the user to modify the following information one item at a time or on a form.
    - Name
    - Gender
    - Birth Date and Place
    - Christening Date and Place
    - Living or Deceased
    - Death Date and Place
    - Burial Date and Place
2. When editing an item, show the prior contributor, date, and reason.
3. Prompt the user to enter a reason for each update.
    - Show the prior reason, if available, in the reason entry box
    - The label of this entry box should be “Why this edit is correct”
    - The user can add to it, write over it or delete it.
    - Provide a button or prompt to "Add Sources"

4. SAVE or CANCEL the edit, and return to the Modify screen.


1
Modify Person's Vital Information
*Select a link to edit the information
User May Select Any Link to Edit or Verify Information
Name
Tyler Johnson
Gender
Male
Birth
2 June 1880
Arlington, Arling
United States
Christening
16 June 1880
Arlington, Arlington, Virginia,
United States
Living/Deceased
Deceased
Death
20 August 1909
Atlanta, Fulton, Georgia,
United States
Burial
23 August 1909
St. Augustus Cemetery, Atlanta,
Fulton, Georgia, United States
DONE
2
Edit Name
Tyler Johnson Jr
4
SAVE
CANCEL
Last Contribution:
October 5, 1980 by Julie Johnson
Reason: Personal knowledge.
Reason this edit is correct:
3
"Jr" was left off the full name as visible on headstone.
Sources:
Add Source
Headstone Image of Tyler Johnson Jr from billiongraves.com
For example, with the Names link selected, the user can edit the name information.

5. Click DONE to return to the previous function in your app.


Modify Person's Vital Information
*Select a link to edit the information
Names
Tyler Johnson Jr
Gender
Male
Birth
2 June 1880
Arlington, Arlington, Virginia, United States
Christening
16 June 1880
Arlington, Arlington, Virginia, United States
Living/Deceased
Deceased
Death
20 August 1909
Atlanta, Fulton, Georgia, United States
Burial
23 August 1909
St. Augustus Cemetery, Atlanta, Fulton, Georgia, United States
5
DONE
When done, return to the previous location in your app

See Also

- Person Data Format (XML)
- Person Data Format (JSON)
- Persons Guide
- Create a Person

Person Delete

A person can be deleted after the relationships to existing persons are deleted.

Typically you should remove relationships rather than delete a person. Rarely would you delete a person you did not add.

Assumptions

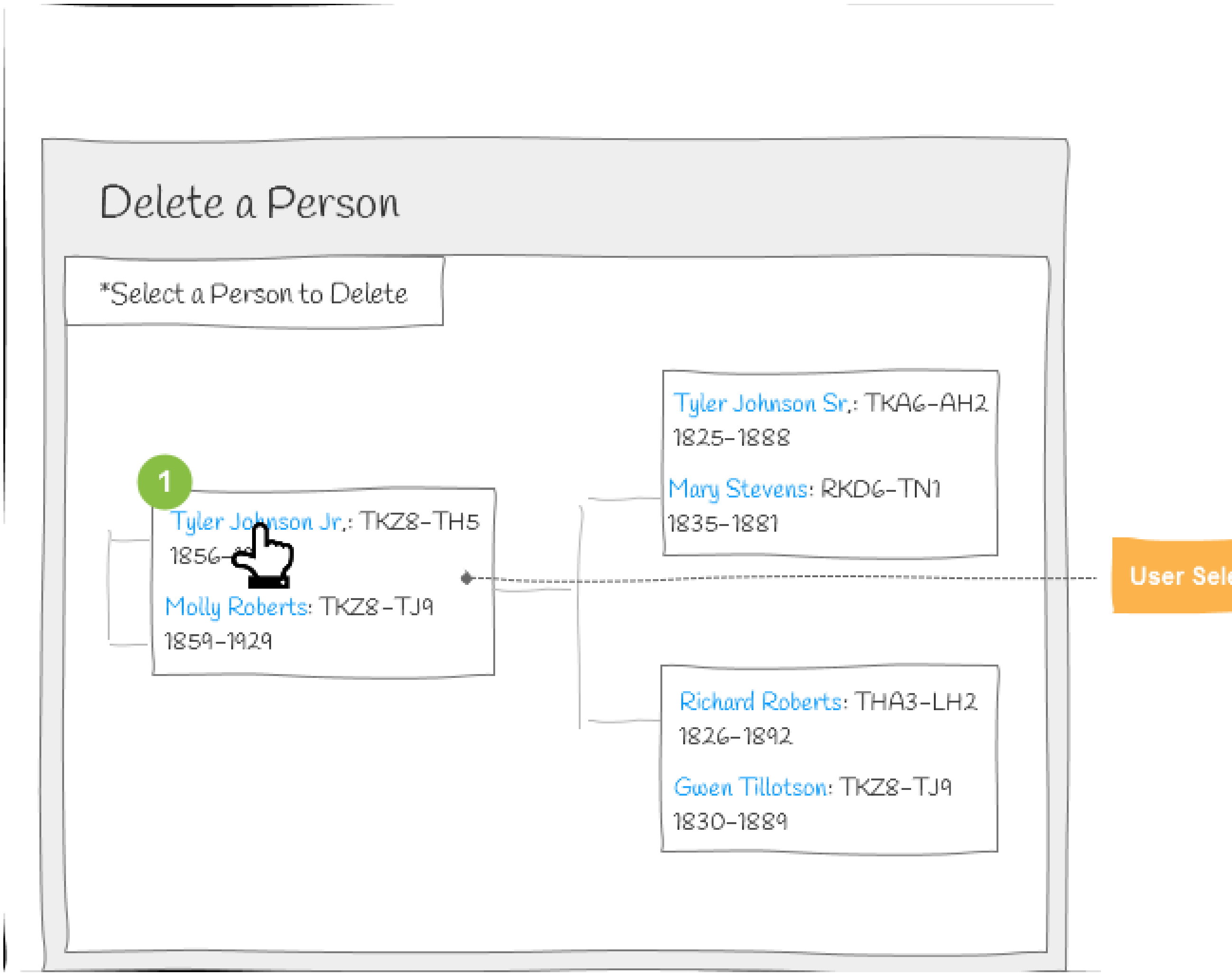
- The user is signed in to FamilySearch through the third-party application.
- The user has searched for matches in Family Tree and found no possible matches.
- The user knows how the new person is related to an existing person in Family Tree.

Programming Steps

1. Search for an existing person and show all information about the person.
2. Show discussions for the person.
3. Show Relationships the person has.
4. Show sources that are attached to the person.
5. Delete relationships.
6. Delete the person.

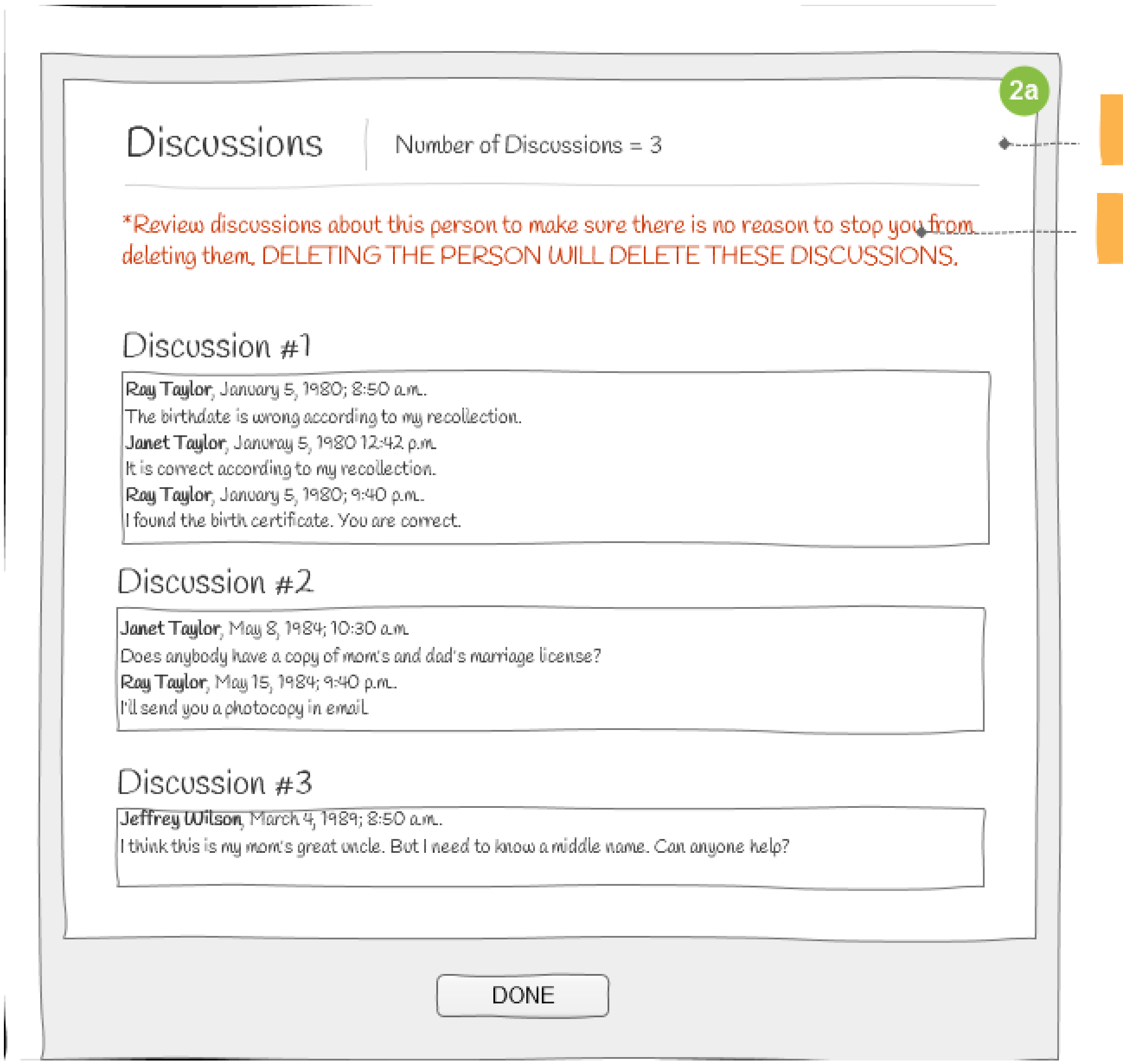
User Interface Suggestions

1. The user selects a person to delete.



2. Display the person's vital information. a) Display the discussions with a cautionary message. b) Display the relationships with a cautionary message. c) Display the sources with a cautionary message.
3. Click PROCEED or CANCEL.


Delete a Person
Tyler Johnson Jr. – TKZ8-TH5
8 June 1856 – 9 March 1924
2
Name
Tyler Johnson Jr.
Gender
Male
Birth
8 June 1856
Arlington, Arlington, Virginia, United States
Christening
12 October 1856
Death
9 March 1924
Atlanta, Fulton, Georgia, United States
Burial
St. Augustus Cemetry, Atlanta, Fulton, Georgia, United States
View Discussions
Number of Discussions = 3
View Relationships
Couple-Relationships = 1
Parent-Child Relationships= 4
View Sources
Number of Sources = 5
PROCEED
3
CANCEL


2a
Discussions
Number of Discussions = 3
*Review discussions about this person to make sure there is no reason to stop you from deleting them. DELETING THE PERSON WILL DELETE THESE DISCUSSIONS.
Discussion #1
Ray Taylor, January 5, 1980; 8:50 a.m.
The birthdate is wrong according to my recollection.
Janet Taylor, Janvray 5, 1980 12:42 p.m.
It is correct according to my recollection.
Ray Taylor, January 5, 1980; 9:40 p.m.
I found the birth certificate. You are correct.
Discussion #2
Janet Taylor, May 8, 1984; 10:30 a.m.
Does anybody have a copy of mom's and dad's marriage license?
Ray Taylor, May 15, 1984; 9:40 p.m.
I'll send you a photocopy in email.
Discussion #3
Jeffrey Wilson, March 4, 1989; 8:50 a.m.
I think this is my mom's great uncle. But I need to know a middle name. Can anyone help?
DONE

2b

Relationships

Couple-Relationships = 1   Parent-Child Relationships= 4

*Review this person's couple and parent-child relationships to determine if the relationships need to be deleted instead of the person

Spouses

Tyler Johnson Jr. – TKZ8-TH5
1856-1924

Molly Roberts – TKZ8-TJ9
1859-1929

Married
29 MAY 1878
Arlington, Virginia, United States

Children

George Tyler Johnson – TKY1-TP2
1880-1909

Timothy Jonathon Johnson – TLY7-TQ4
1883-1940

Sophia Johnson – TZH6-LP5
1885-1948

Buford Johnson – TPW9-KS1
1887-1955

Parents

Tyler Johnson Sr. – TKA6-AH2
1825-1888

Mary Lois Stevens – RKD6-TN1
1835-1881

Married
2 MAY 1853
Oklahoma City, Oklahoma,
Oklahoma, United States

Siblings

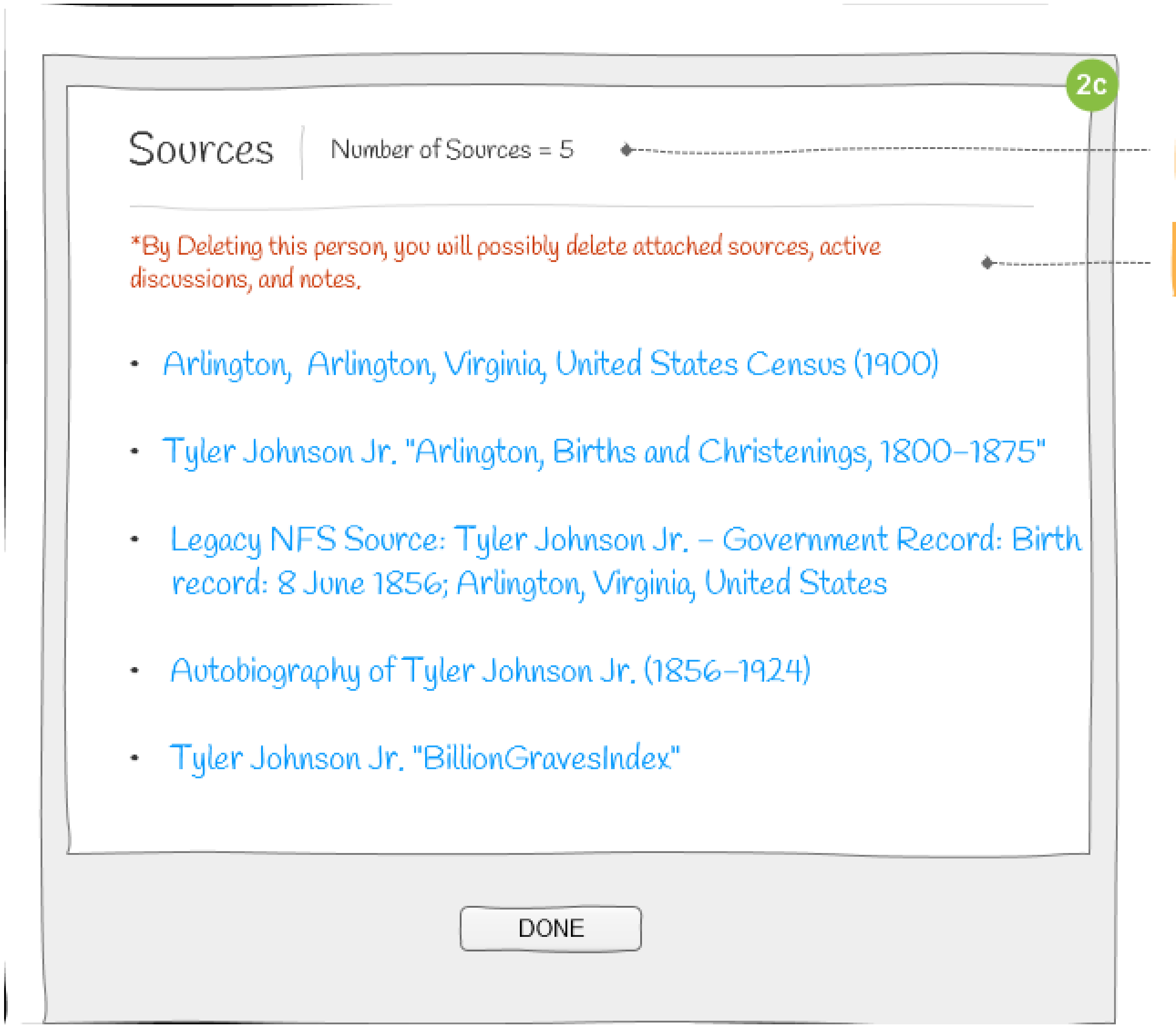
Sarah Mary Johnson – PTY5-HP2
1855-1910

Tyler Johnson Jr. – TKZ8-TH5
1856-1924

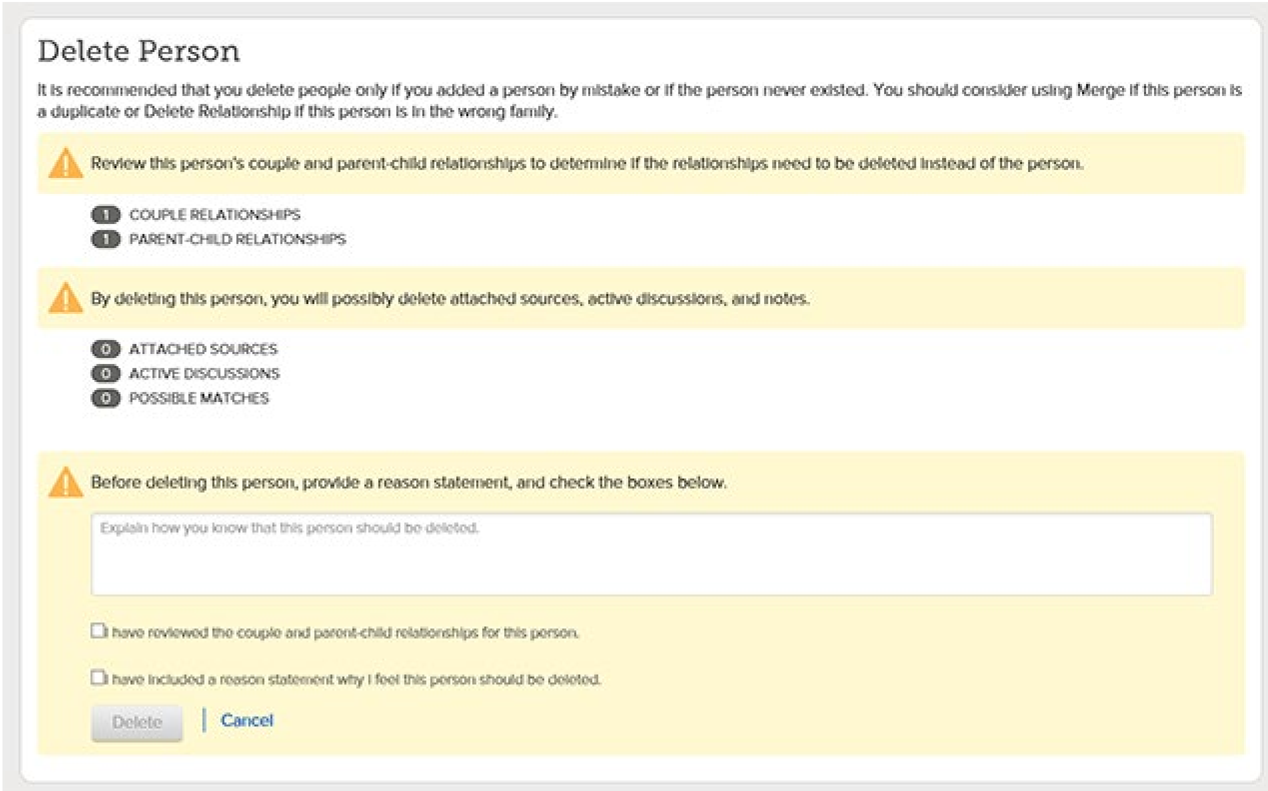
Jeanie Johnson – YGB8-FS2
1858-1928

John Dunford Johnson – TQT7-VS4
1860-1938

DONE



4. Present a confirmation page, and prompt the user to provide a reason for deleting this person.
5. Prompt the user to agree to the following three statements:
   - I have read the other reason statements above.
   - I have reviewed the couple and parent-child relationships for this person.
   - I have provided a reason statement regarding why I feel this person should be deleted.
6. Click DELETE the person or CANCEL.



Standardization of Event Information

Data for vital events such as birth, marriage, and death should be standardized using normalized values from [FamilySearch Authority APIs]() or the partner standards. FamilySearch Authority APIs are the preferred method of standardization. Standardized values help to maintain accuracy in identifying person matches and duplicates.

Places should be standardized for the following:

- Events in a person's life
- Couple relationships events
- Parent-child relationships events

Other information fields should be standardized as well but are not required.

Assumptions

- The user must be signed in to FamilySearch through the app.
- The user is entering a date or place for an event in a person’s life.

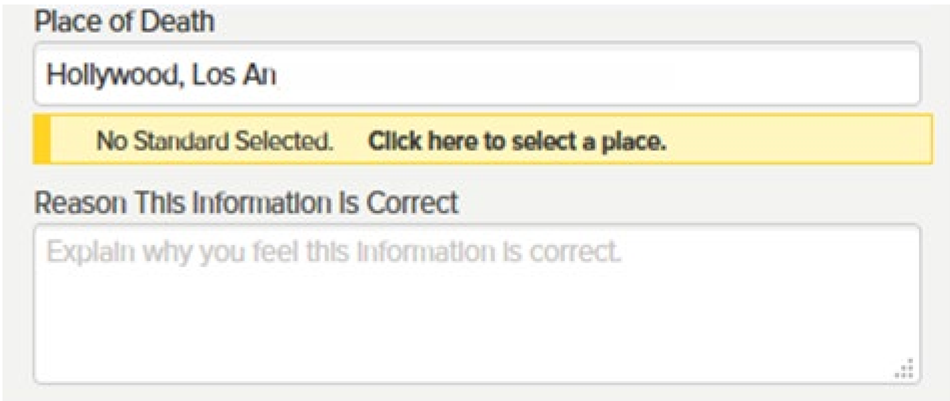
Programming Steps

1. Capture information using a dropdown form item.

2. Search possible standards from the FamilySearch Authorities API module or the partner's authorities database.
3. Populate the dropdown with possible standardized values to select.
4. Update the website with standardized value.

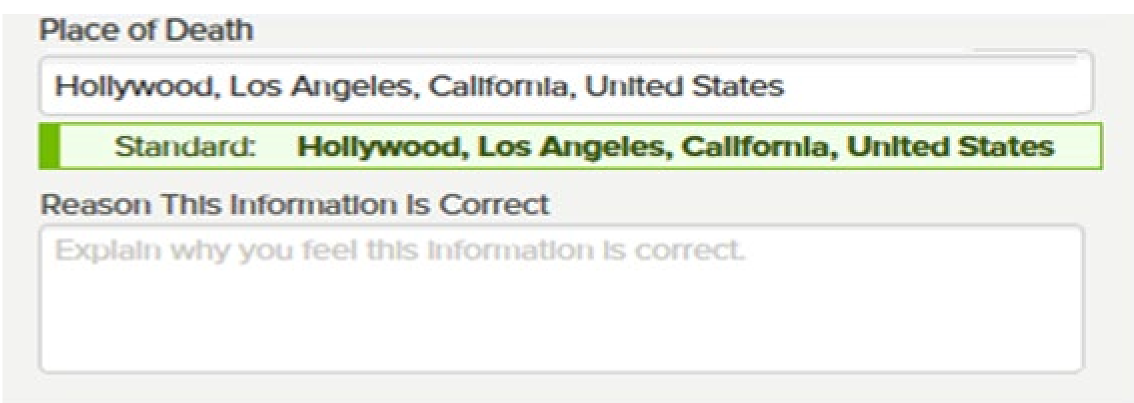
User Interface Suggestions

1. Prompt the user to enter place and request a standardized value



2. Provide selection of standardized choices

3. Indicate that value selected is standardized



See Also

- Seach For Places

Person Compare and Transfer of Partner Application Data to FamilySearch Data

Person vital facts from third-party databases can be compared with values in FamilySearch. Values can be added or replaced in FamilySearch as long as the existing attribution and prior contributor's reason have been viewed. All transfers to FamilySearch must include the new reason for the addition or change.

Assumptions

- Select a person in the partner application to compare with a person in FamilySearch that is already linked.

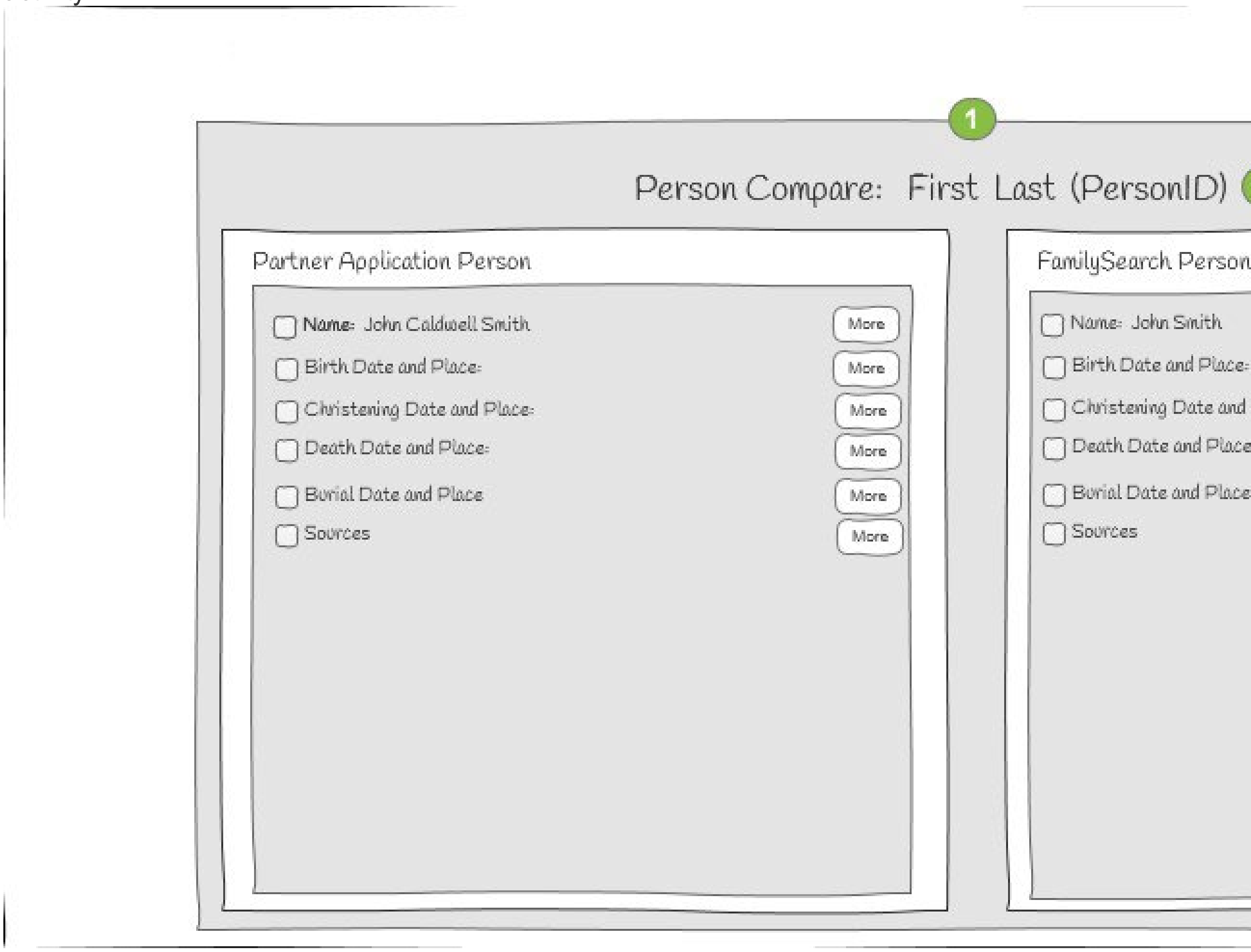
Programming Steps

1. Read a Person.
2. Update a Person.
3. Add a Person.
4. Add a Fact (conclusion).

User Interface Suggestions

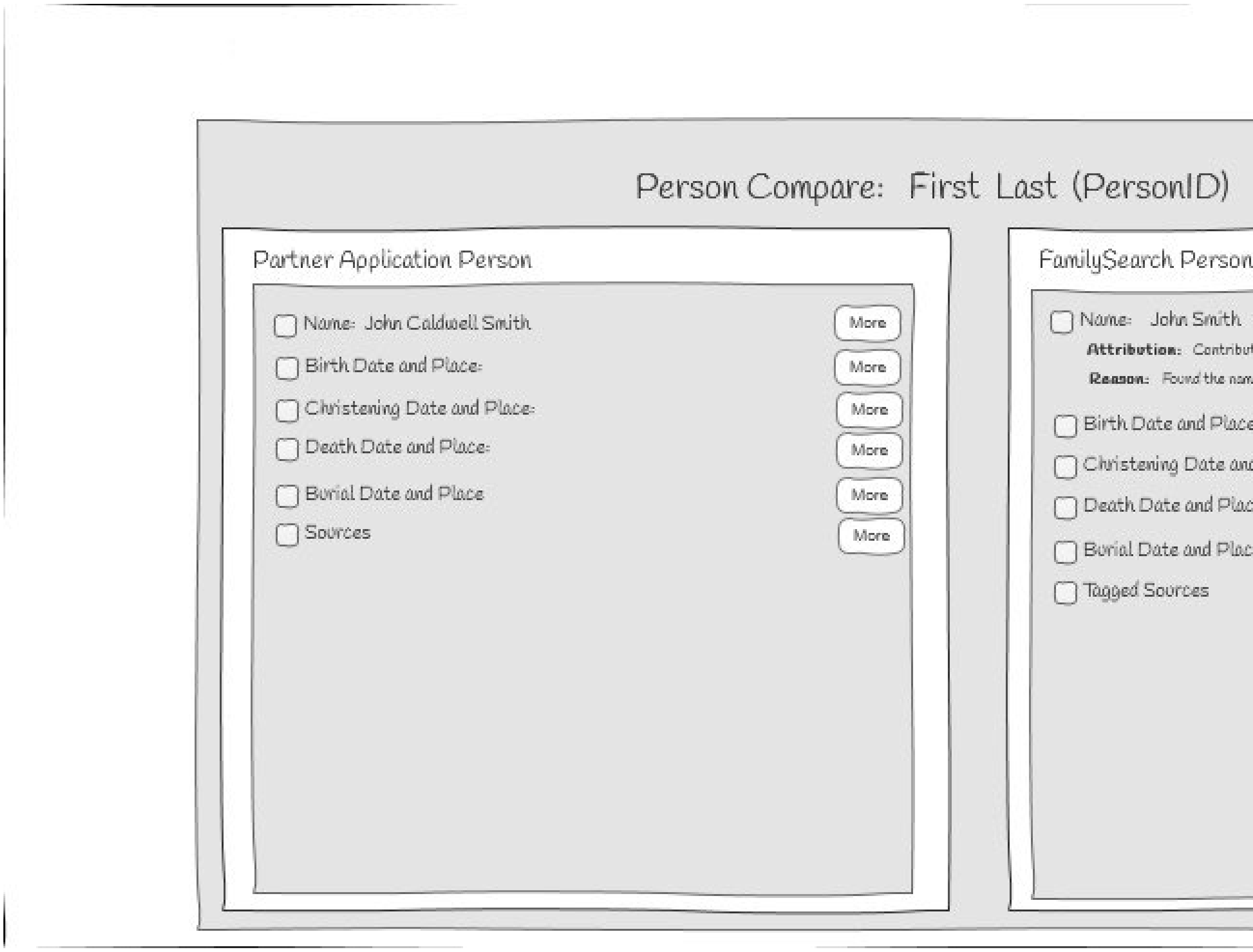
1. Show a comparison of person information between the partner application and FamilySearch.
2. Provide buttons or links for Sources, Discussion, and Change History. Sources and Discussions should show a count of

activity.



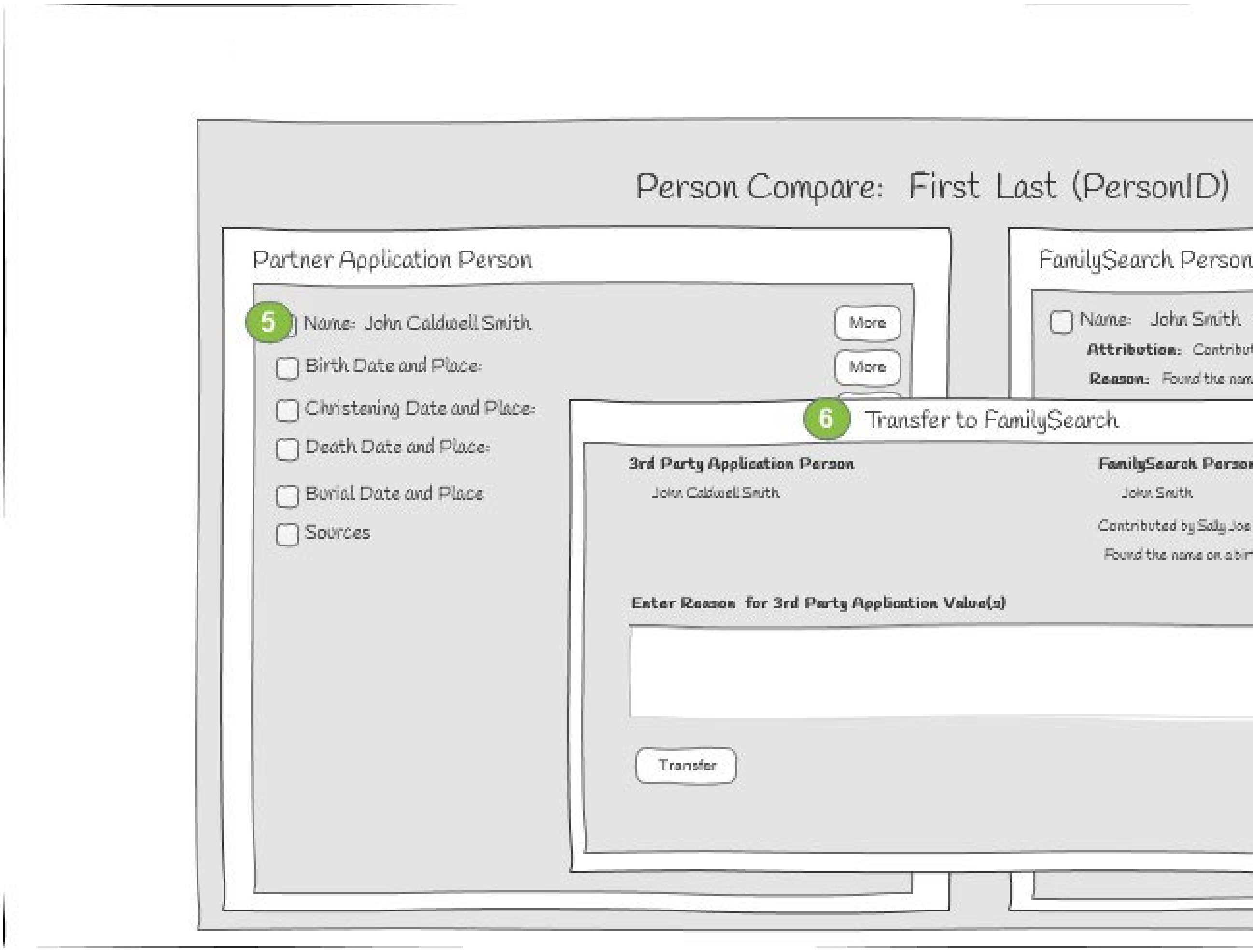
3. More and Less buttons show and hide additional details of fact or events listed for the person. More should always include the attribution and the contributor's reason. The application could choose to show More as the default. The attribution and reason must be shown before any change is made to the FamilySearch person.
4. The tagged sources can link to a list of sources or display the sources in FamilySearch for this person with the facts or events that are tagged. Optionally, the process may be used for Notes and/or Person

Sketches.



5. Marking a checkbox indicates that the third-party application's values for the fact or event should be added to or the values replaced in FamilySearch.
6. Present a dialog that provides the detail information with a listing of the sources. The user can submit or cancel the change after entering the reason for the change.
   The user is returned to the comparision page where the changed values now

match.



Notes:

- The user flow does not depict a confirmation of changes before committing the changes. You might want to present a summary of changes and ask for confirmation before writing the changes.
- Group multi-value facts (nonvitals) need to be compared and maintained individually. All FamilySearch values need to be presented underneath the fact. For example, each job and date needs to be presented for the occupation multi-value fact.
- The attribution and prior reasons need to be presented and the ability to add "why this edit is correct" for each value.
    - Show the prior reason, if available, in the reason entry box
    - The label of this entry box should be “Why this edit is correct”
    - The user can add to it, write over it or delete it.

Check for Duplicates

Preventing the performance of duplicate persons and their information is one of Family Tree's main purposes.

Issues

To prevent duplication, a person's record must be merged with the duplicate person that contains the duplicated information. This may create the following challenges:

- End users may become discouraged or bored if they have to review too many possible duplicates.
- Users may be confused if they merge some duplicate records and then later find that more possible duplicates exist for the same person.

Minimum Requirements

- Offer the user through a prompt, menu item, or button the option to check for "Possible Duplicates". This option should appear before or on the screen where users add or change the person's information.
- Provide your own duplicate maintenance using the API, or redirect the user to the FamilySearch Possible Duplicates page specified in the "duplicates" context parameter. For more information, see the Persistent Identifiers guide.

Restore to a Previous State in FamilySearch

The change history log in FamilySearch can be used to assist in performing the following changes.

- Delete a resource that was added
- Restore a resource that was deleted
- Restore a person deletion

Deleted persons, their relationships, and the reasons for changes are found in the Change History Log. Restoring a deleted person restores the person and the parental relationship.

Note: Restore functionality is not required for partner application certification.

Programming Steps

1. Select a person to review.
2. Read the change history, and display it.
    - For actions, list the value, attribution, and reason for the current and historic actions.
    - For persons, list the person and the parent information.
3. The user selects an item to restore and enters a reason.
4. Restore the change that was selected.

- Restore an action.
- Restore a person.
- Restore a parental relationship.

User Interface Suggestions

When selecting the person to restore, display the following information:

- Summary information of the deleted person and the parents.
- The current value and the value of the data being restored.
- The reason, contributor, and last modified date for the current value.
- Give an opportunity to enter a reason for performing this restore.

Change or Delete Relationships in FamilySearch

Couple relationships and parent-child relationships can be changed or deleted. Before the relationship is changed or deleted, show the people in the relationship, the date of the relationship, the type of relationship, the attribution, and the prior reason and sources. Before the change is committed, prompt the user to enter a reason for the change.

Programming Steps

Update Couples

1. Read Couple Relationship of a person to obtain the "Couple ID".
2. Read Couple Relationship Sources to view details of the sources attached to a couple.
3. Read Couple Relationship Source Reference to view a list of sources attached to the couple.
4. Update Persons of a Couple Relationship to change the persons in the couple.
5. Update Couple Relationship Conclusions to add or change statistics relevant to a couple.

Update Parent-Child

1. Read Child-Parent Relationship
2. Read Child-Parent Relationship Sources
3. Read Child-Parent Relationship Source Reference
4. Update Child-Parent Relationship
5. Update Child-Parent Relationship Conclusions

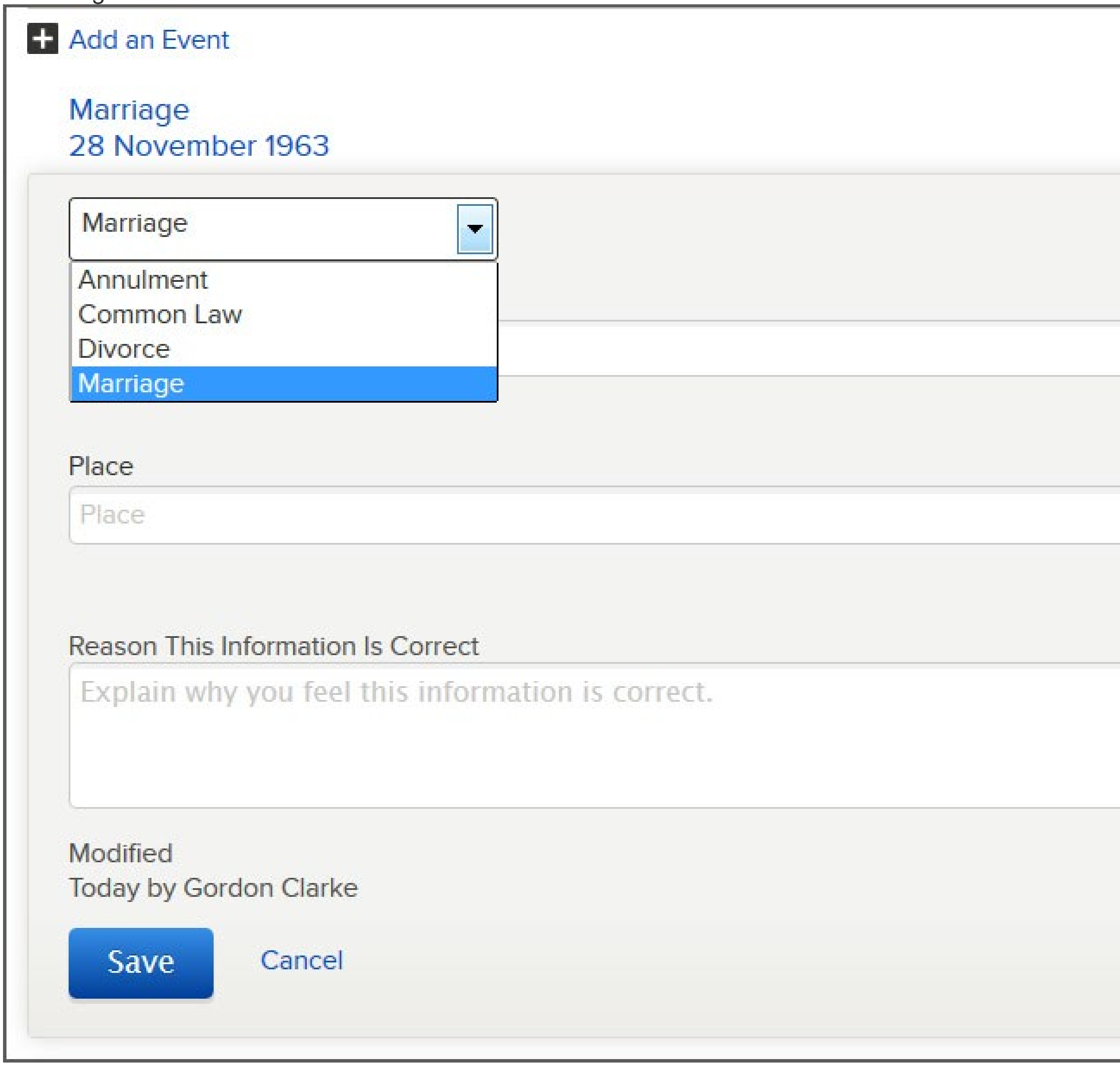
Delete Couples and Parent-Child

1. Delete Couple Relationship
2. Delete Couple Relationship Conclusion

User Interface Suggestions

1. The user selects a person.
2. Show the existing relationships.
3. The user selects a relationship to change or delete.
4. Change Relationship

   Couple Relationships are changed by adding an event for marriage, annulment, common law, or divorce. Show the existing relationship event and date. Select the relationship, enter the date, place, and the reason before saving or canceling.

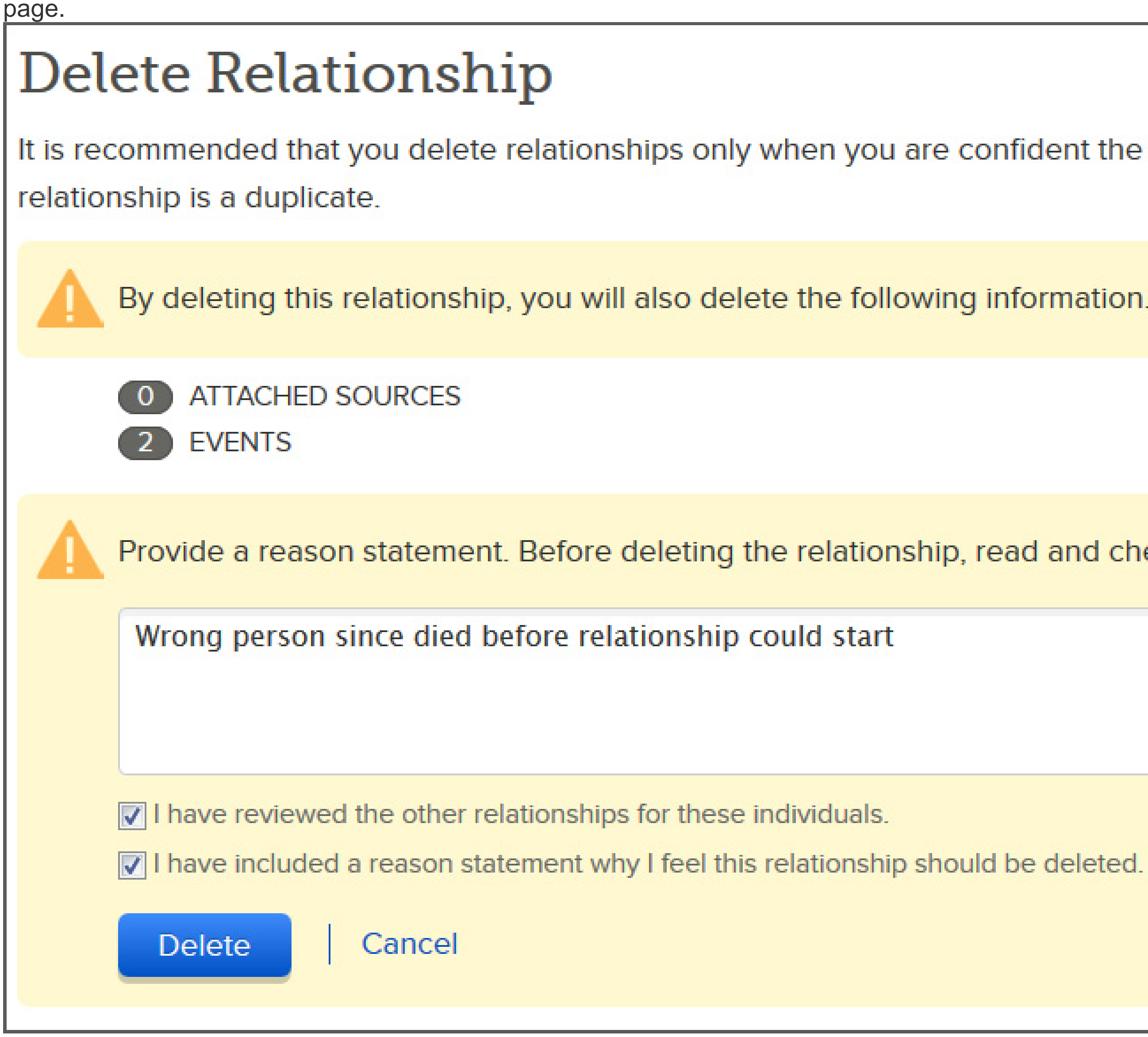


Parent Relationships are changed by editing the parents of a child relationship type which includes adoptive, biological, guardianship, foster, or step. Select the relationship, enter the date, place, and the reason before saving or canceling. The relationship can be set separately for each parent.

Child Relationships are changed by selecting the child and changing the relationship with the parents.

5. Delete Relationship

Show the vital facts of the persons involved in the relationship and the number of attached sources and events. Prompt for a reason for the deletion before deleting, or cancel and return to the Select Relationships page.



Merge Persons Maintenance in FamilySearch

After person records in the application database have been linked to FamilySearch persons, possible duplicate records must be identified and merged where appropriate.

Check for duplicates, and provide enough information to determine if the person should be merged. Present a name-by-name comparison so the user can make a correct decision.

Assumptions

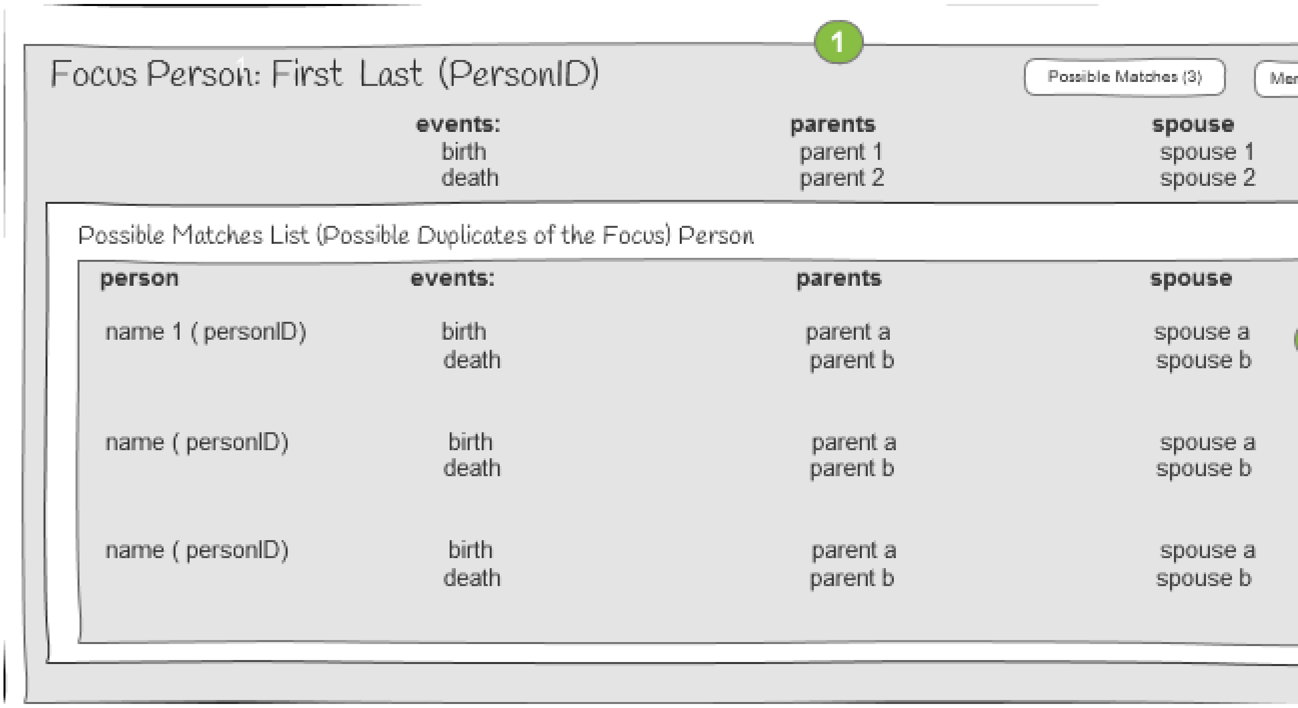
- Possible matches have been requested for a focus person from that person's summary or detail page.

Progamming Steps

1. [Read Person Possible Duplicates]
2. [Merge Persons]

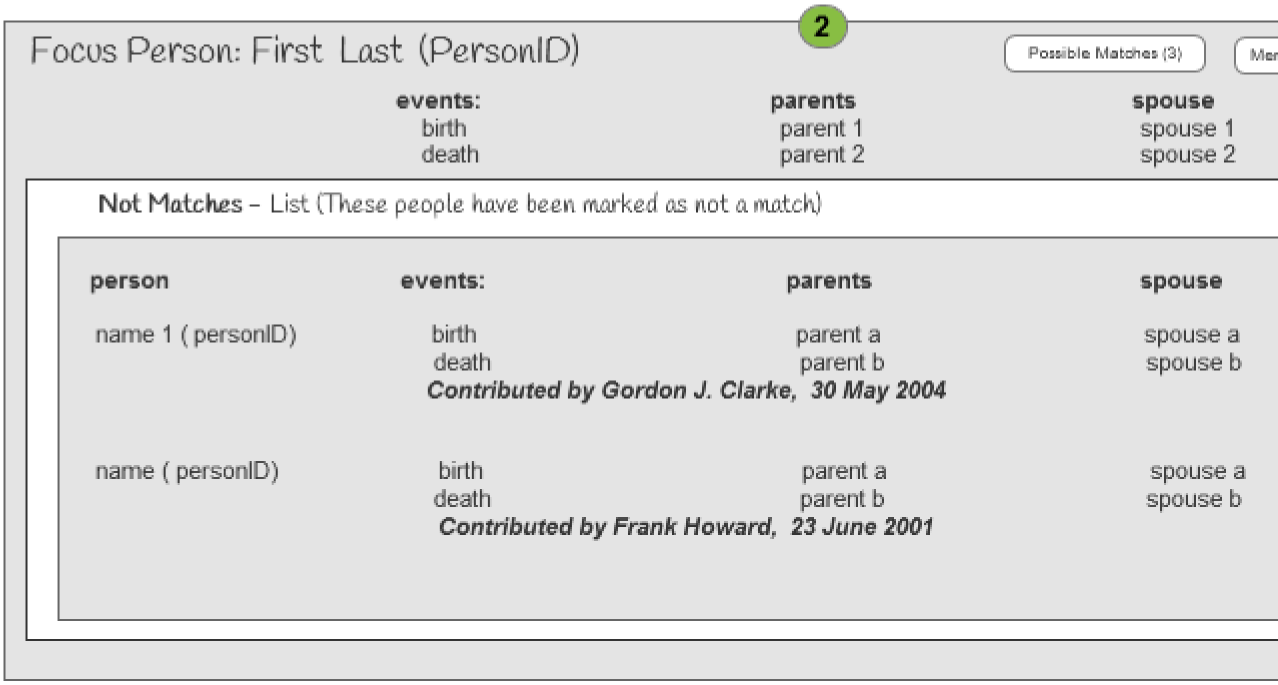
User Interface Suggestions

1. Display a list of the possible matches of the selected or focus person. Include name, person ID, birth and death events, parents, spouse, and sources. Provide a "Merge by ID" option to merge a specific person ID with the focus person.
2. Provide an option to show a list of the persons who are not matches.
3. For each possible match, provide a preview merge option which will show a side-by-side comparison of the person records with an opportunity to merge them.



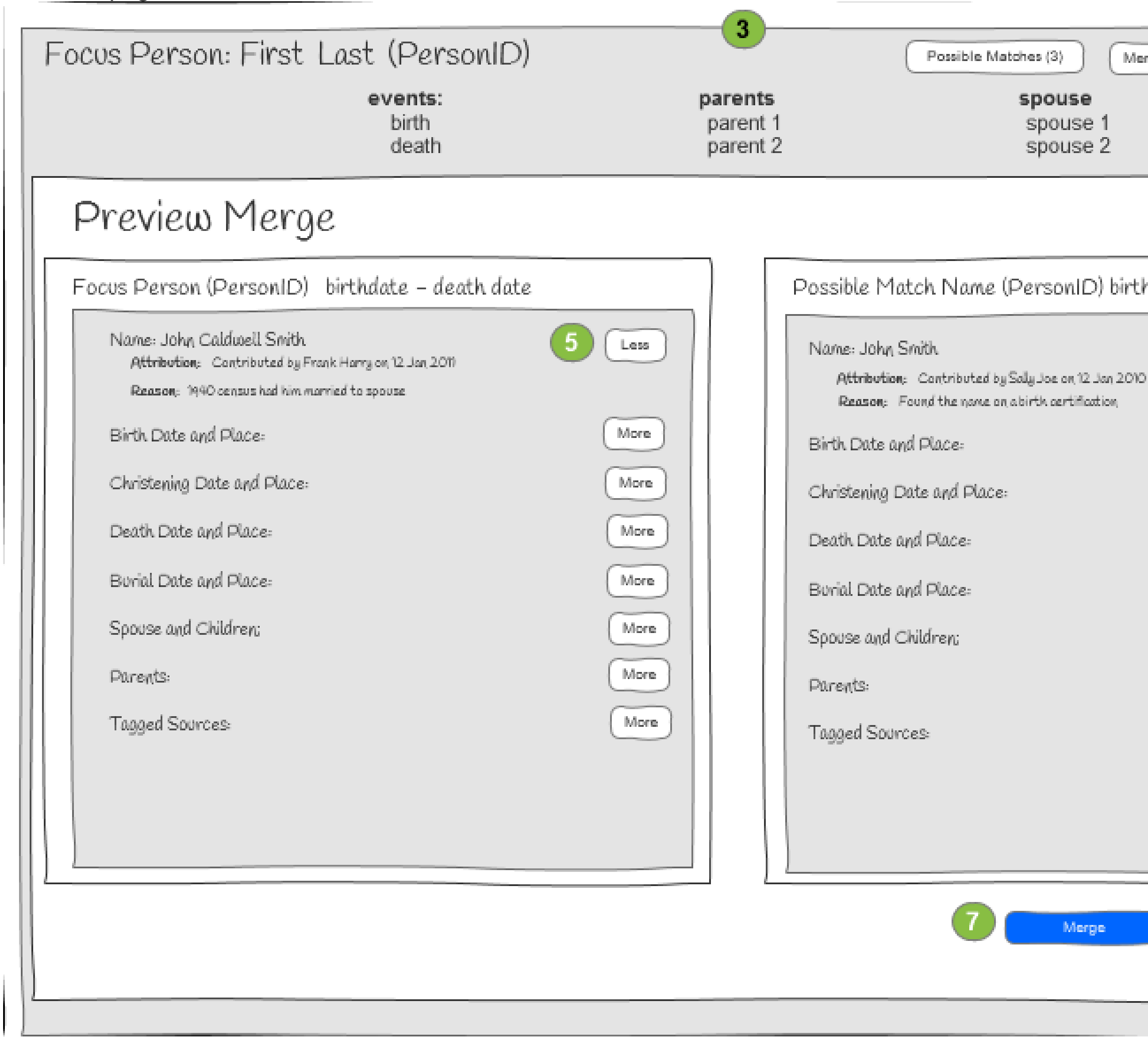
4. The ***Not Matches*** list shows the persons that are specified as not a match with the focus person. "Review Details" goes to the ***Review Not a***

Match page.



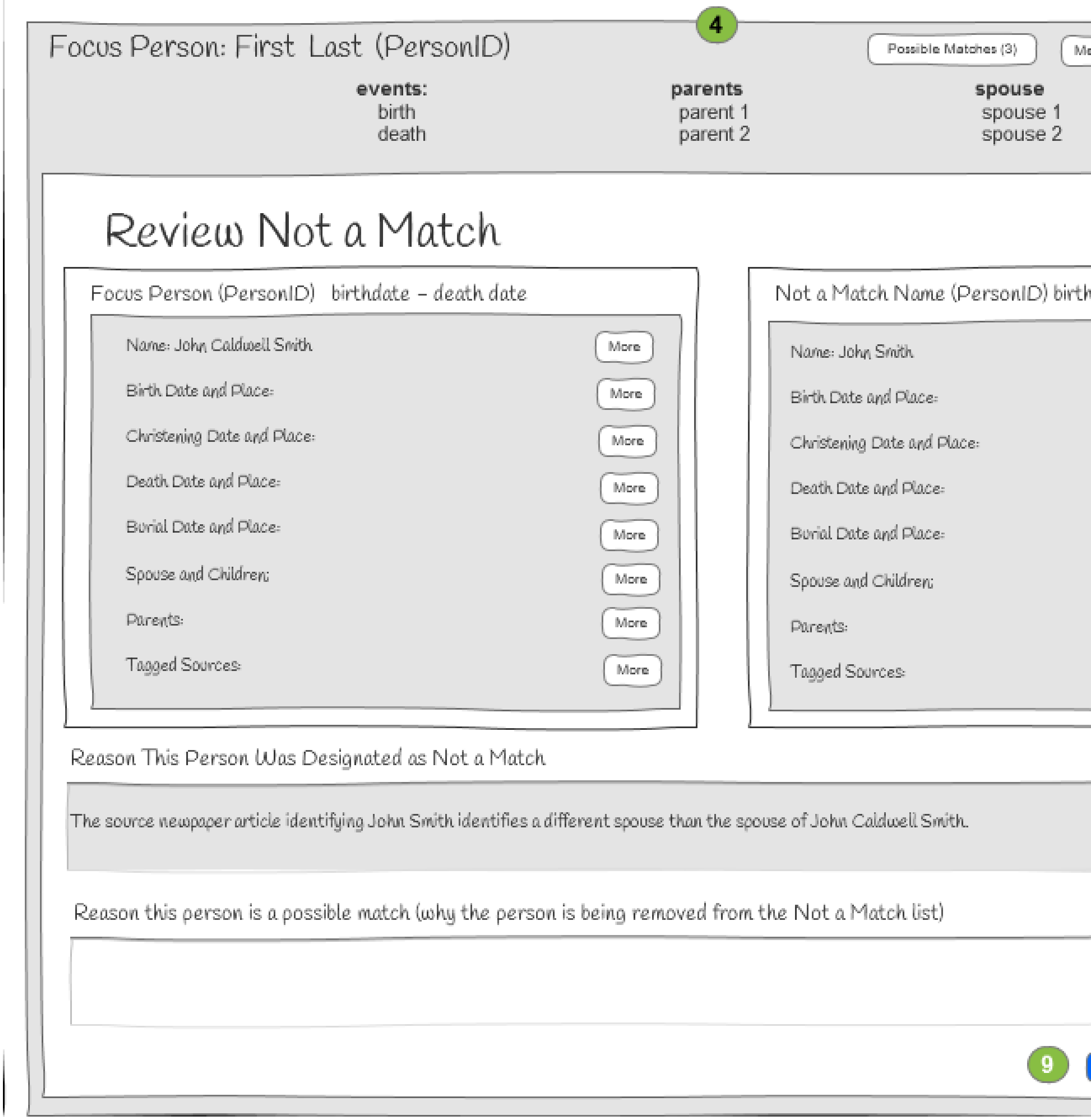
5. On the ***Preview Merge*** page, the expansion of possible match information reveals a choice to replace the focus person information.
6. Information will not be merged unless this radio button is selected.
7. Merge goes to a confirmation page of the matches that will be merged.
8. Not a Match goes to the ***Not a Match Dialog***. Cancel makes no changes and returns to the ***Possible***

Matches page.



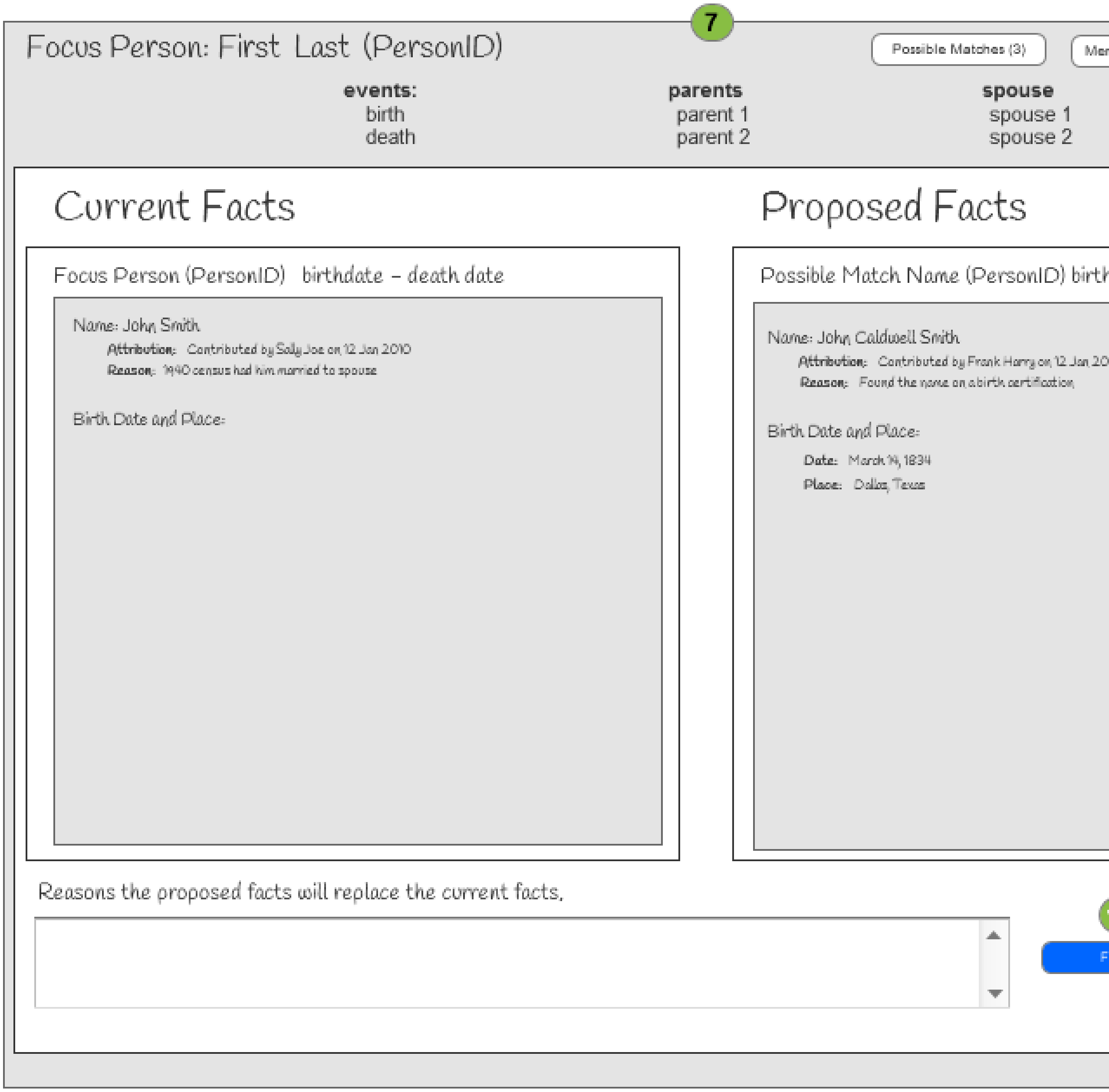
9. The ***Review Not a Match*** page displays the focus person on one side and the not a match person on the other side of a page. Prompt the user for a reason. Remove removes the person from the not a match list and returns to the ***Possible Matches*** page. Cancel returns to the ***Possible***

Matches page.



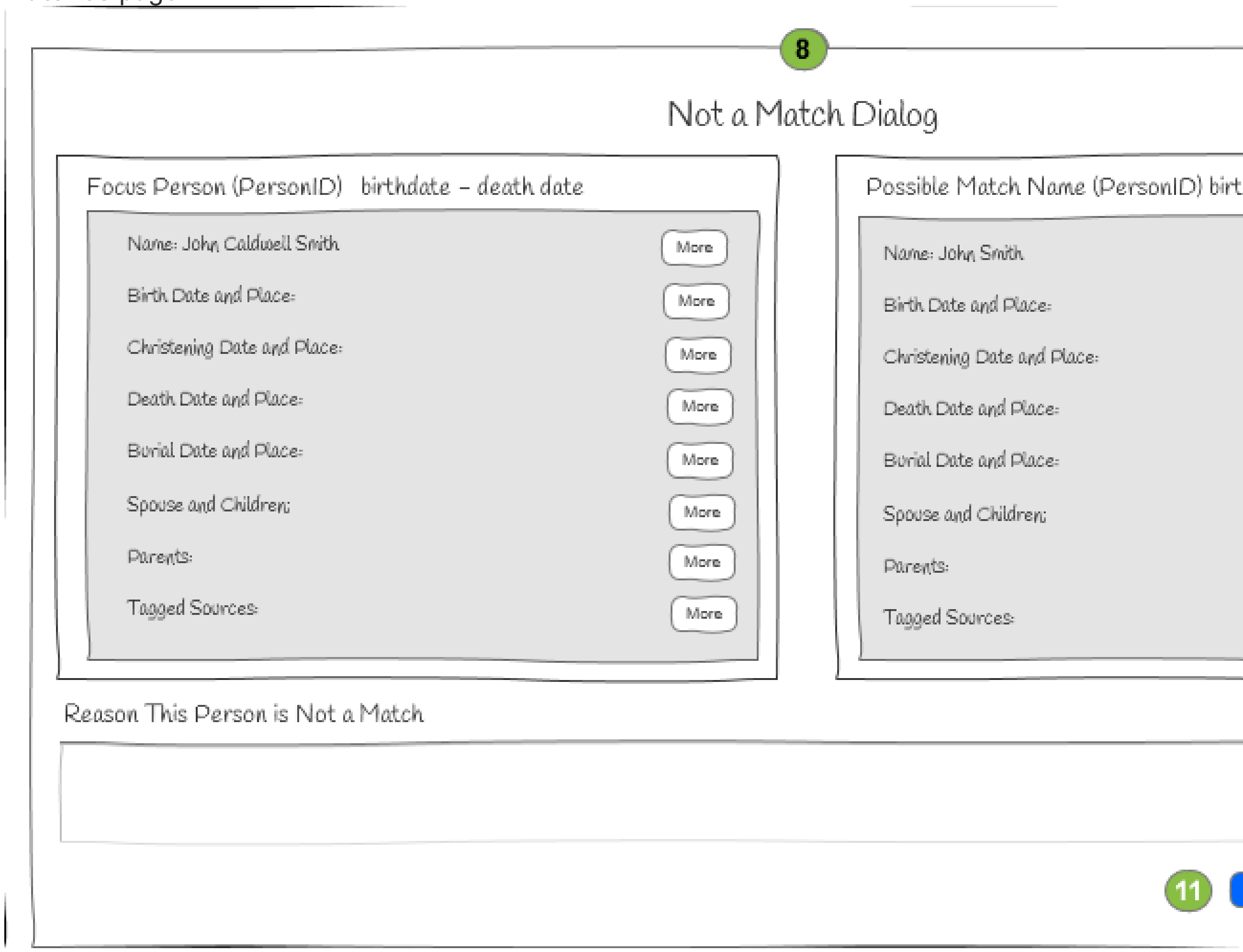
10. The ***Merge confirmation page*** shows the facts that will be merged. Prompt for a reason. Finish completes the merge and returns to the ***Possible Matches*** page. Cancel returns to the ***Possible***

Matches page.



11. The ***Not a Match Dialogue*** page displays the focus person on one side and the possible match person on the other side of a page. Prompt the user for a reason. Finish puts the person on the not a match List and returns to the ***Possible Matches*** page. Cancel returns to the ***Possible***

Matches page.



Important!

- The application must make it clear to the user which information will survive. The option to submit a reason why the information is correct must be offered.
- The application must provide the ability to select which specific values will be replaced and which will be preserved after the merge, including vitals and other information, parents, and children.
- The user must be able to choose to bring over any relationships and sources that do not exist with the focus person.
- The application must prompt for a reason this merge is correct before finishing the merge.
- Summary of any sources related to the person should be displayed before finalizing the merge.

Restore Merged Persons

Sometimes two persons are incorrectly merged and need to be unmerged. That is, the merged persons need to be restored as two individuals. The history of merged persons is found in the Change History of the surviving person. After the merge is found in the Change History, the persons can easily be restored to their state before the merge.

Assumptions

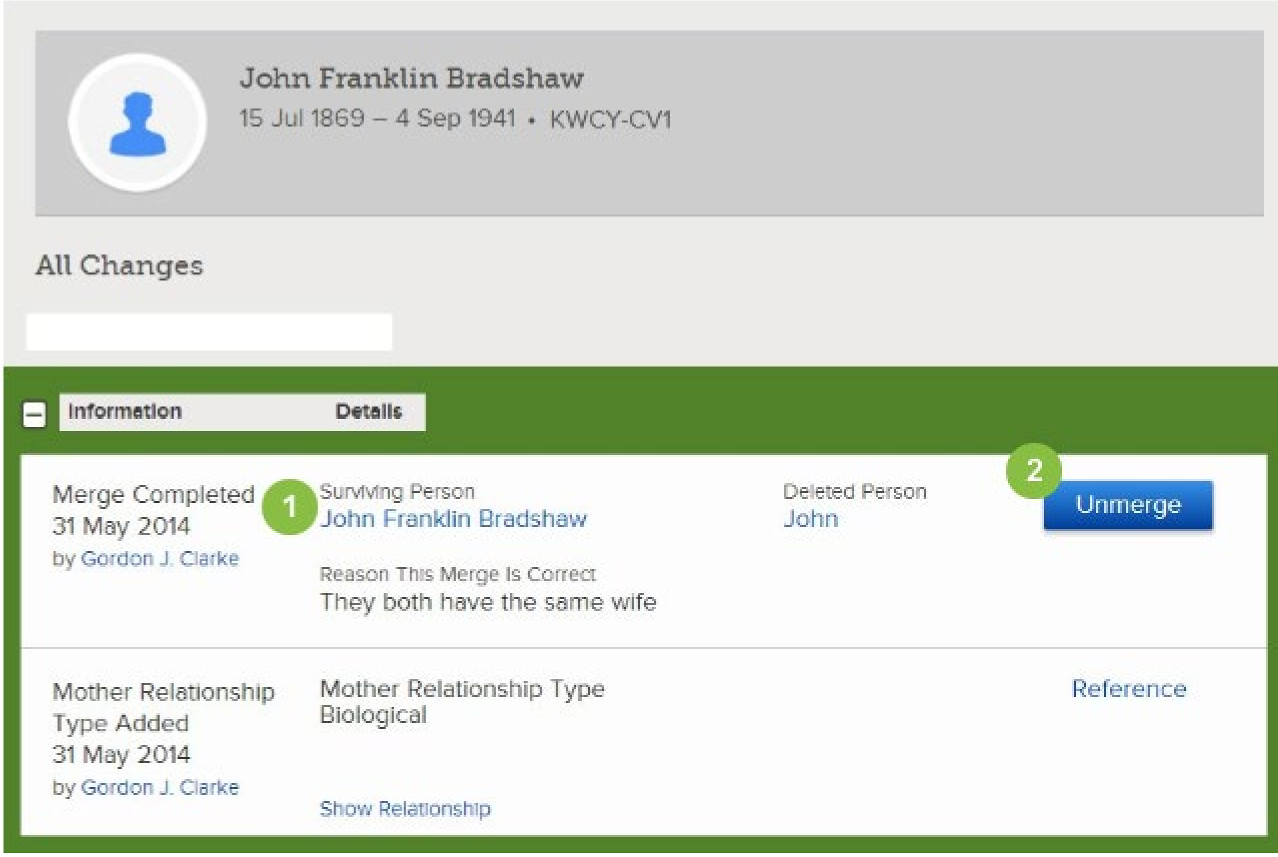
- You have the Person ID of the person you want to unmerge.

Programming Steps

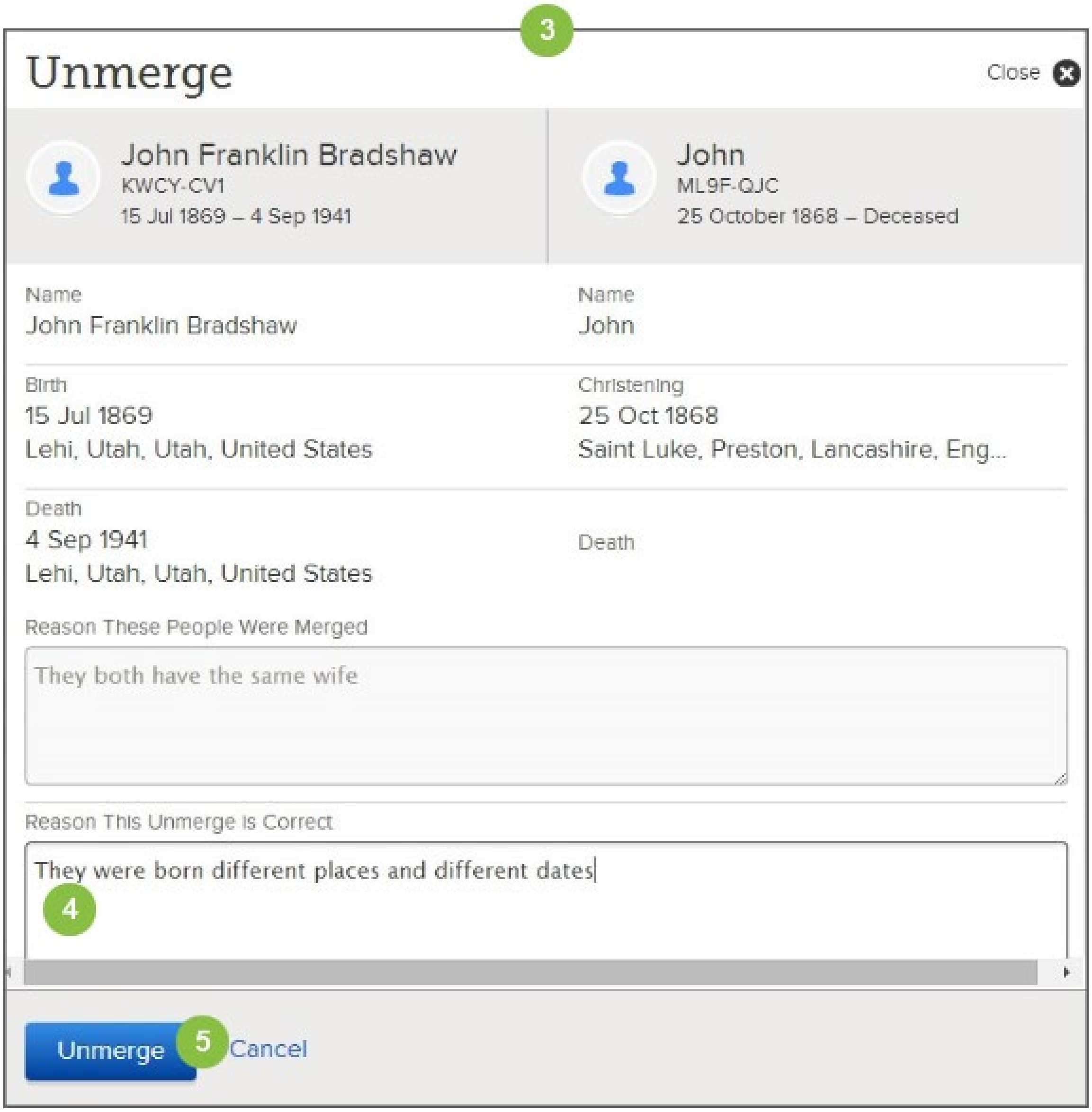
1. Display the Change History of the merged person.
2. Display the summary of the two persons that were merged together and the reason they were merged.
3. Unmerge the selected person.

User Interface Suggestion

1. View the Change History of a person, and locate the merge that you want to unmerge.
2. Proceed to unmerge the data.



3. View the summary of the two people to be unmerged and the reason they were merged.
4. Enter a reason to unmerge.
5. Finalize the unmerge or cancel.



See Also

Merging Guide

Notify Users of Watched Person Changes and Other Activities

FamilySearch Family Tree has the ability to watch and unwatch changes on a person, and to show a list of watched persons. The user who sets the watch on a person gets an email notification about each change. This allows Family Tree to easily keep users aware of changes to persons they are interested in.

FamilySearch does not offer the Watch feature in a public API. Therefore, application developers must assume the responsibility to notify their users of changes to persons they are interested in. Before important tasks are performed, the users should be notified of changes that have happened. Alternatively, the application could display the most recent changes in the change history every time the user views person summary or person detail information. Activity stats should also be reported for Sources and Discussions changes.

Programming Steps

- Compare FamilySearch data with application stored data.
  - Person data. Read Person
  - Etag value. Read Person Change Summary
  - Recent changes to Change History. Read Change History
- Additional Sample Code for Activity Stats.
  - Read Person Sources
  - Read Person Source References
  - Read Discussion References

User Interface Suggestion

Compare with Stored Person and Etag Values

1. The user requests to see person information.
2. Report the number of sources and discussions on the person.
3. Show possible differences between the partner data and FamilySearch data.
4. Prompt the user to do a detailed comparison and decide to update person data on the partner app.

Report on recent changes to Change History

1. The user requests to see person information.
2. Report the number of sources and discussions on the person.
3. List a summary of the latest changes on the request person.
4. Prompt the user to do a detailed comparison of changes and decide to update person data on the partner app.