All I Want [for Christmas]

Cristhian Ulloa, Dirk Stahlecker, Kristin Asmus, Tricia Divita

Design Doc

Motivation

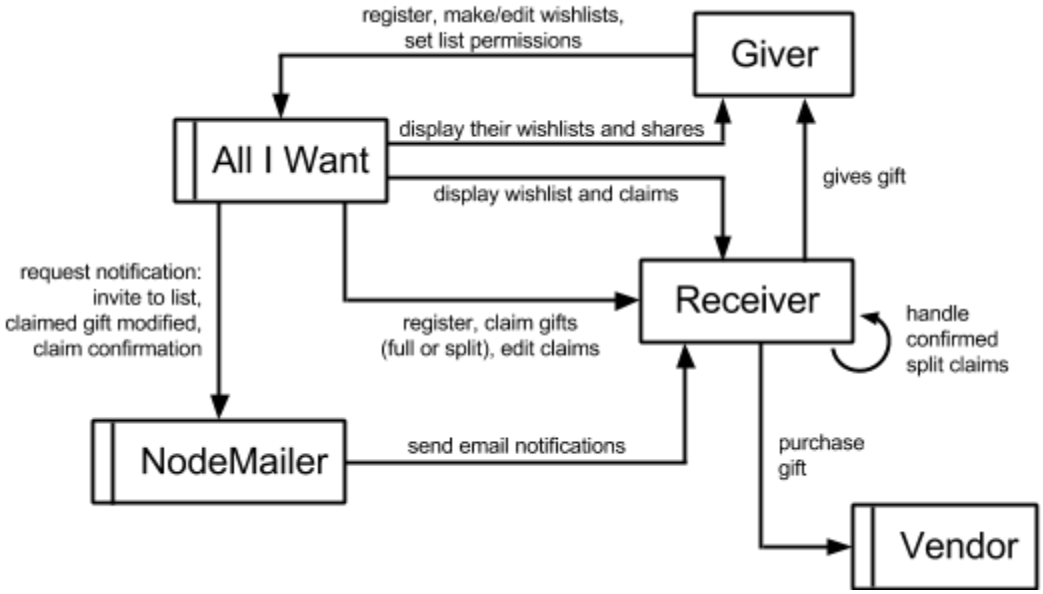
All I Want [for Christmas] is an online app with the goal of simplifying and optimizing gift giving. Its creation is motivated by the fact that gift-giving is often a disorderly task, and many people do not have a system to handle it well.

The purposes are as follows:

1. **Get the gifts you want.** Users create wishlists of gifts they want, and share them with their friends or make them public, so that everyone knows what they would like to receive.
2. **Give only gifts you know they'll like.** Other users can view wishlists and choose to give one or more of the desired gifts.
3. **Avoid duplicate gifts.** Users can make claims on gifts in a wishlist, so gift givers can view what gifts have already been claimed, and avoid getting the same gift for a user.

There are a few aspects that existing solutions do not allow for.
CheckedTwice is one existing site that has much of our functionality, with a few significant exceptions. in CheckedTwice, each user only has a single wishlist, and gifts have the option of being shared differently. But there is no option for sharing with the public, or even with specific users. All gifts are either private or shared with friends and family.
Other more widely known options, like *Amazon Wishlist* or *store registries*, only allow for gifting of specific items from their associated vendors. Our solution will enable users to link to specific items on the web or include descriptive info for a product from a specific store, but will also offer them the freedom of describing what they want without having a specific vendor's product in mind.
Another major point is that there do not seem to be any solutions that allow for the splitting of gifts between multiple users, which will make our app unique in its functionality.

Context Diagram



Concepts

Wishlist: A list of gifts a user is requesting. A wishlist can be shared with other users by its creator, so that they can view it to help them decide what to give the wishlist owner.
(Motivated by purposes 1 and 2.)

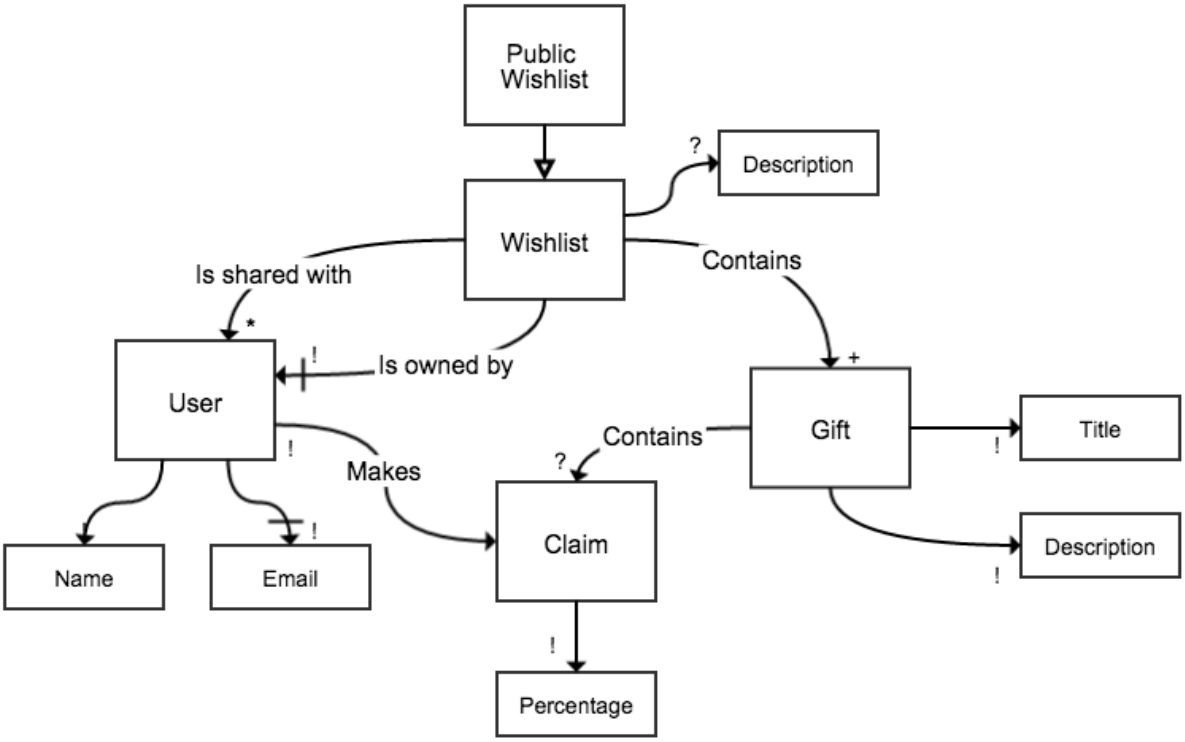
Gift: An item a user wishes to receive. Each gift belongs to a wishlist of a specific user.
(Motivated by purpose 1.)

Claim: A promise that one user will give a wishlist gift to its requester. Users can only claim gifts from wishlists that have been shared with them. (Motivated by purpose 3.)

Split Claim: Like a claim, but the claimant specifies the percentage of the gift that they are willing to cover.
(Motivated by purpose 3.)

User: An individual interacting with the site while logged in with a specific account. Each user can create their own wishlists, and control who they are shared with and what gifts are on them. A user can also view wishlists that others have shared with them and claim (or split a claim for) gifts from those lists. When giving gifts, a user is also responsible for buying the gifts they have claimed and getting them to the requester, or arranging that with those they have split it with.
(Motivated by purposes 1 and 2.)

Data Model



Gift can only be claimed by users who can view the wishlist containing that gift

Private wishlists can only be viewed by users that the list is shared with

Public wishlists can be viewed by anyone, but can also be shared with specific users

Users can't claim gifts on a wishlist they own

Security Concerns

Key Security Requirements:
Because our app involves only claiming gifts and not directly purchasing them, our security concerns are considerably less critical. We want to be able to authenticate users mainly for the purpose of being able to share wishlists with other users with the assurance that only the intended people can actually see it. Users will sign up with an email, name, and password, and may be required to verify their email in order to authenticate. This account information will be hidden from others and encrypted as necessary, with the exception of their name.

Mitigating Standard Attacks:

- **Hoax public wishlists posing as charities.** Because we allow anyone to make an account and create a public wishlist that anyone can contribute to, it is definitely

possible that a user could create a public wishlist posing as a charity for their own personal benefit. However, because our site only enables users to claim gifts, they must deliver gifts on their own. There are two likely scenarios for this - a user could hand-deliver the gift, or mail a package. If they hand-deliver it, they can likely judge for themselves the trustworthiness of the receiving organization. If they mail it, the owner of the public wishlist must include an address or contact info in the description, which conscientious users could verify independently. To encourage this practice, our site will include notes on public wishlists encouraging creators to include information to prove their validity, including a verifiable contact number, address, and/or website. Likewise, we will include a note warning users that All I Want does not verify the owners of public wishlists.

- **Grinches (especially on public pages).** One possible abuse of the claiming system is that a malicious user (a “grinch”) could claim items from a wishlist with the intent of never giving them, and have their claims discourage other users from giving those gifts as well. This is mitigated on private wishlists by the fact that the owner can choose who is able to view and therefore claim items from them. However, the worst case scenario would be a grinch on a public wishlist who arrives uninvited but unimpeded, and claims all of the gifts so that other users who might have contributed give nothing. We’re still considering how best to combat this, but one potential solution would be to allow public wishlists to disable claims completely (especially for wishlists where there’s no need to limit the number of each gift received). Another solution could be to limit the number of claims a user can make on each public wishlist.
- **Hackers attempting to extract user information (emails and passwords).** We will encrypt emails and passwords in our internal database in order to mitigate this threat.
- **Malicious data entry.** We will clean all user inputs to attempt to prevent injection attacks.

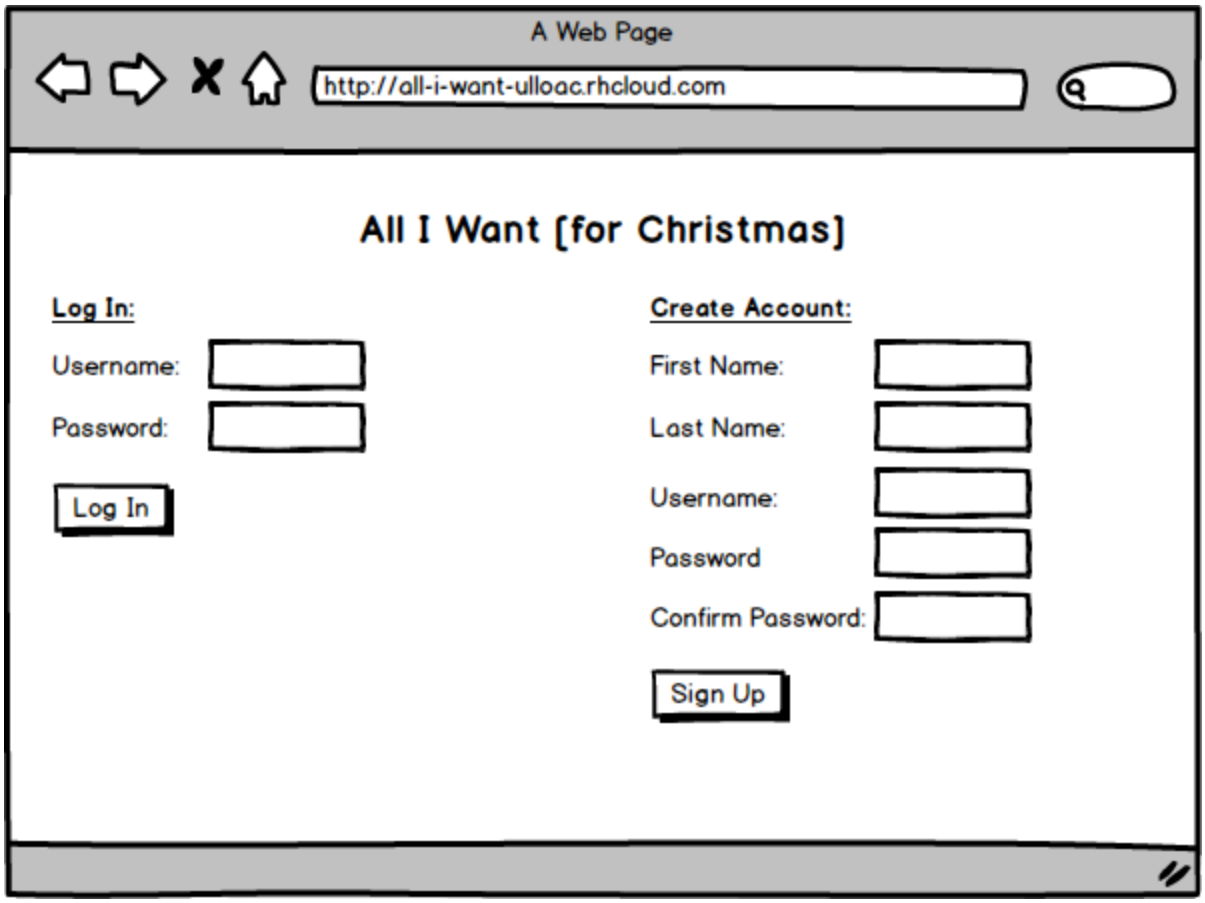
Threat Model

- Can assume no major criminal attacks due to the small scale of the project and the minimal information stored (email).
- Hoaxes are less likely given that our project doesn’t involve actually purchasing and sending items. However, this is still a threat that should be given some consideration.
- There isn’t much motivation for grinches, but the internet is full of trolls so we should do our best to mitigate the negative effects they could cause for other users.

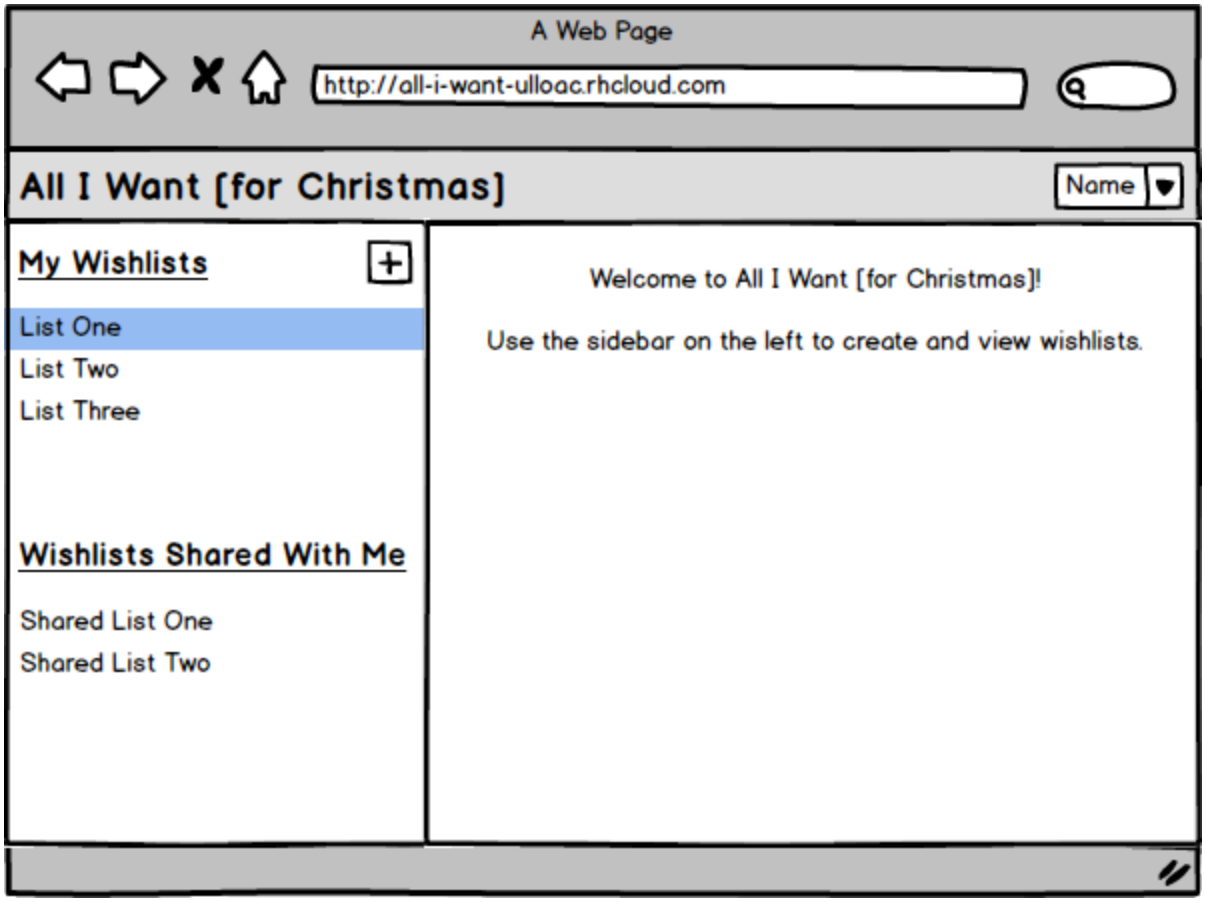
User Interface

Wireframes

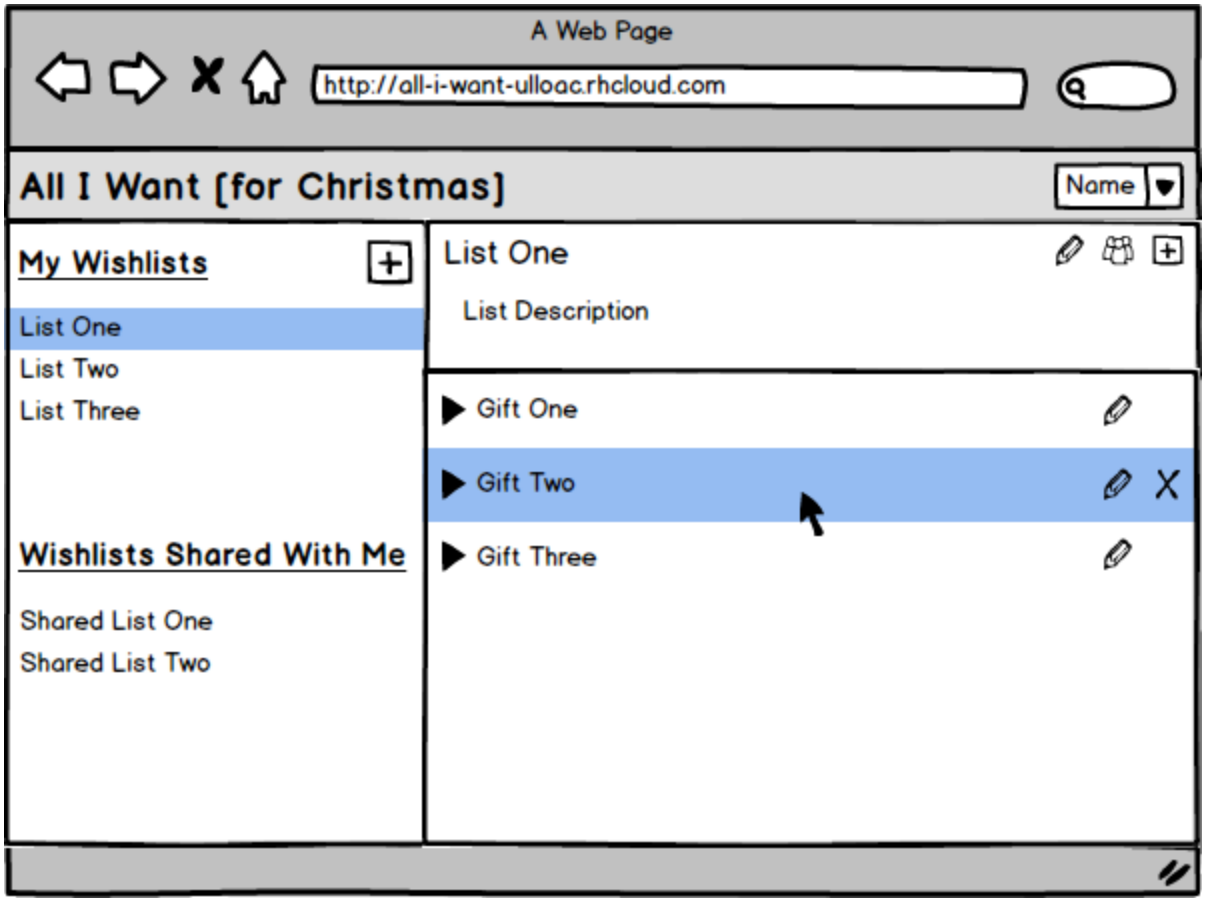
Login Page:



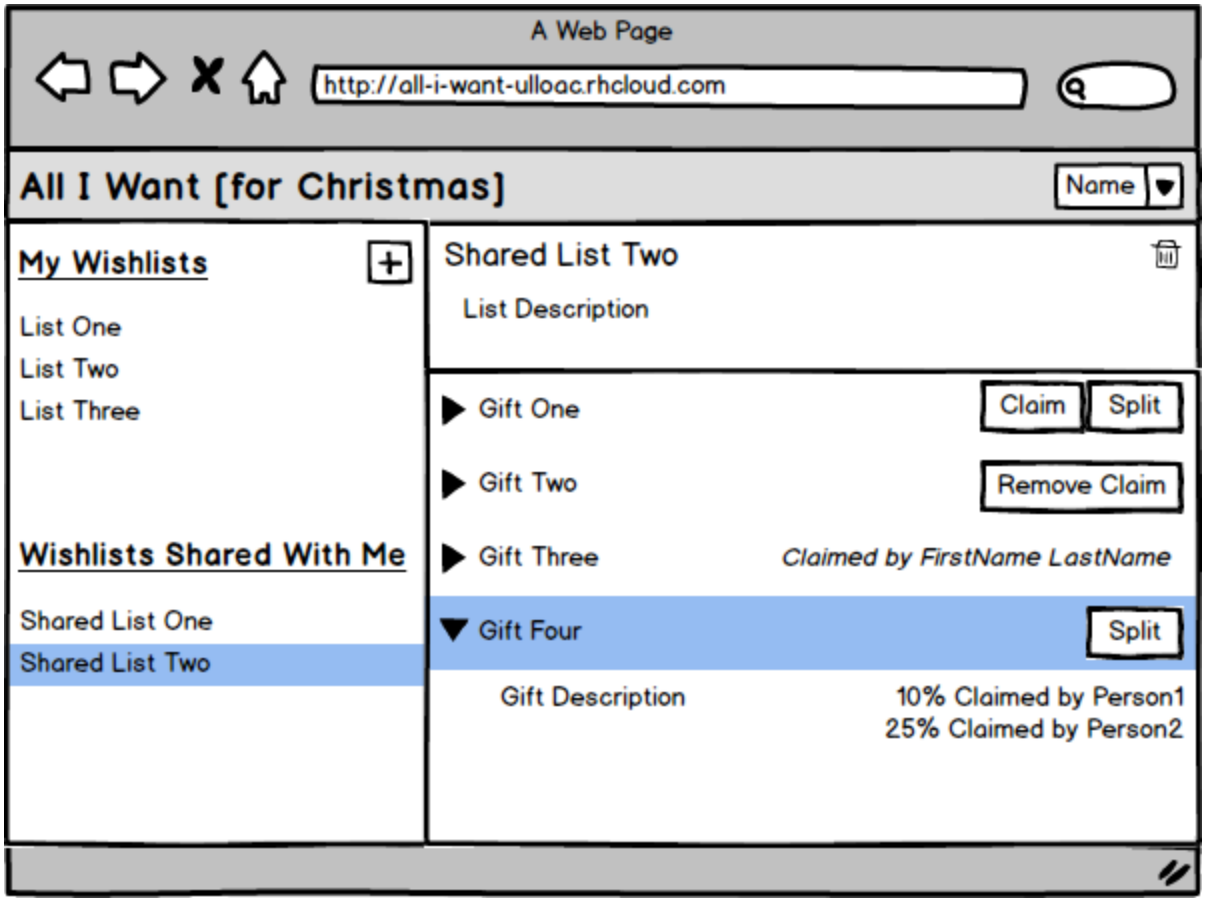
Landing Page:



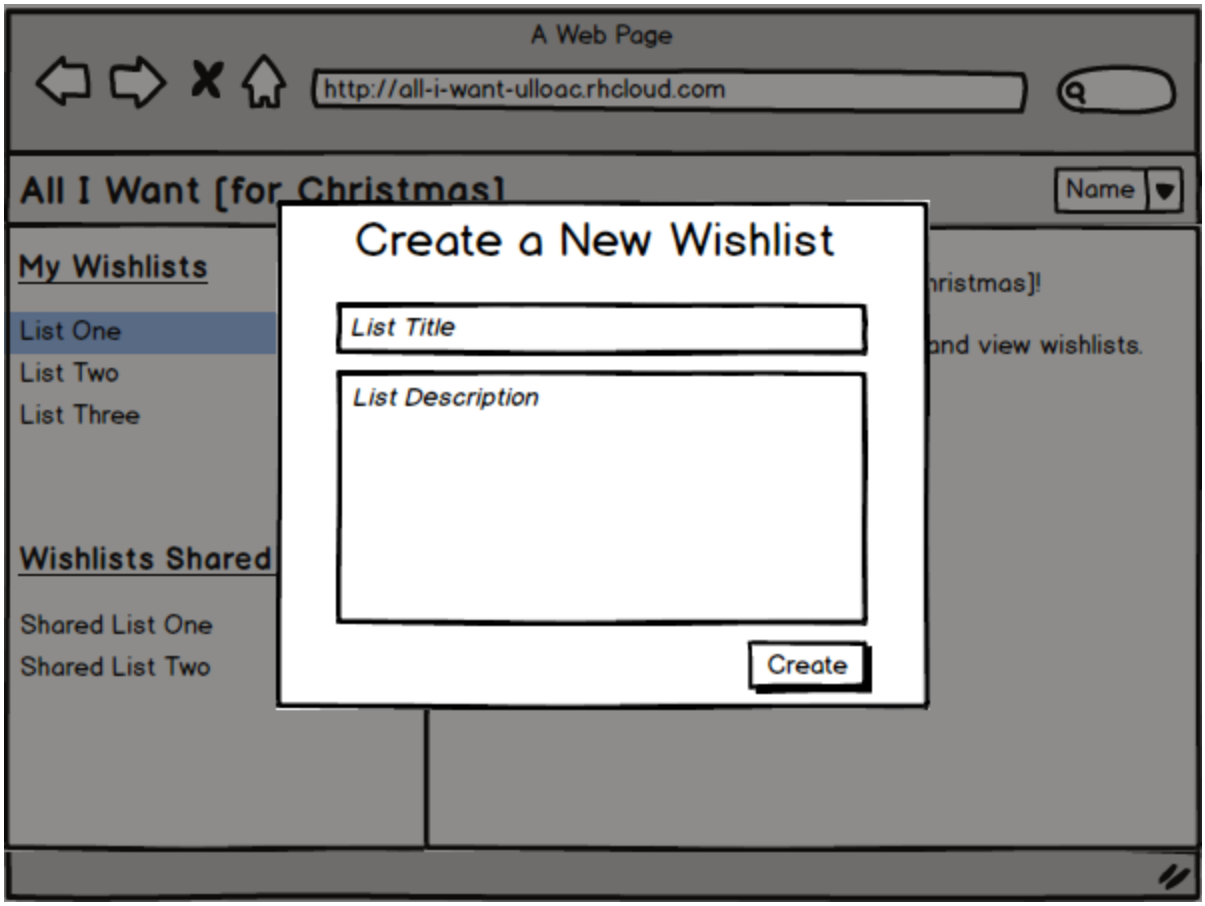
My Wishlist Page:



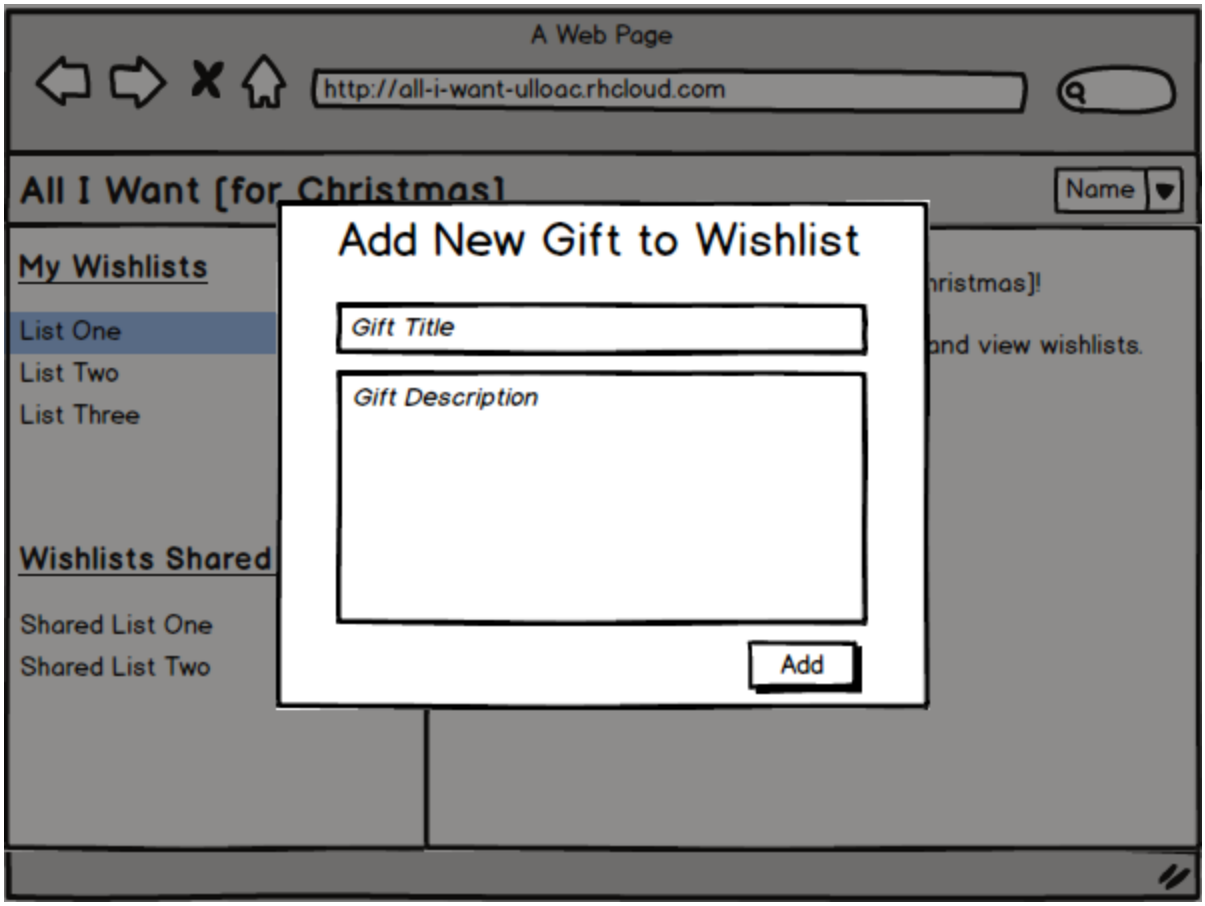
Wishlist Shared with Me Page:



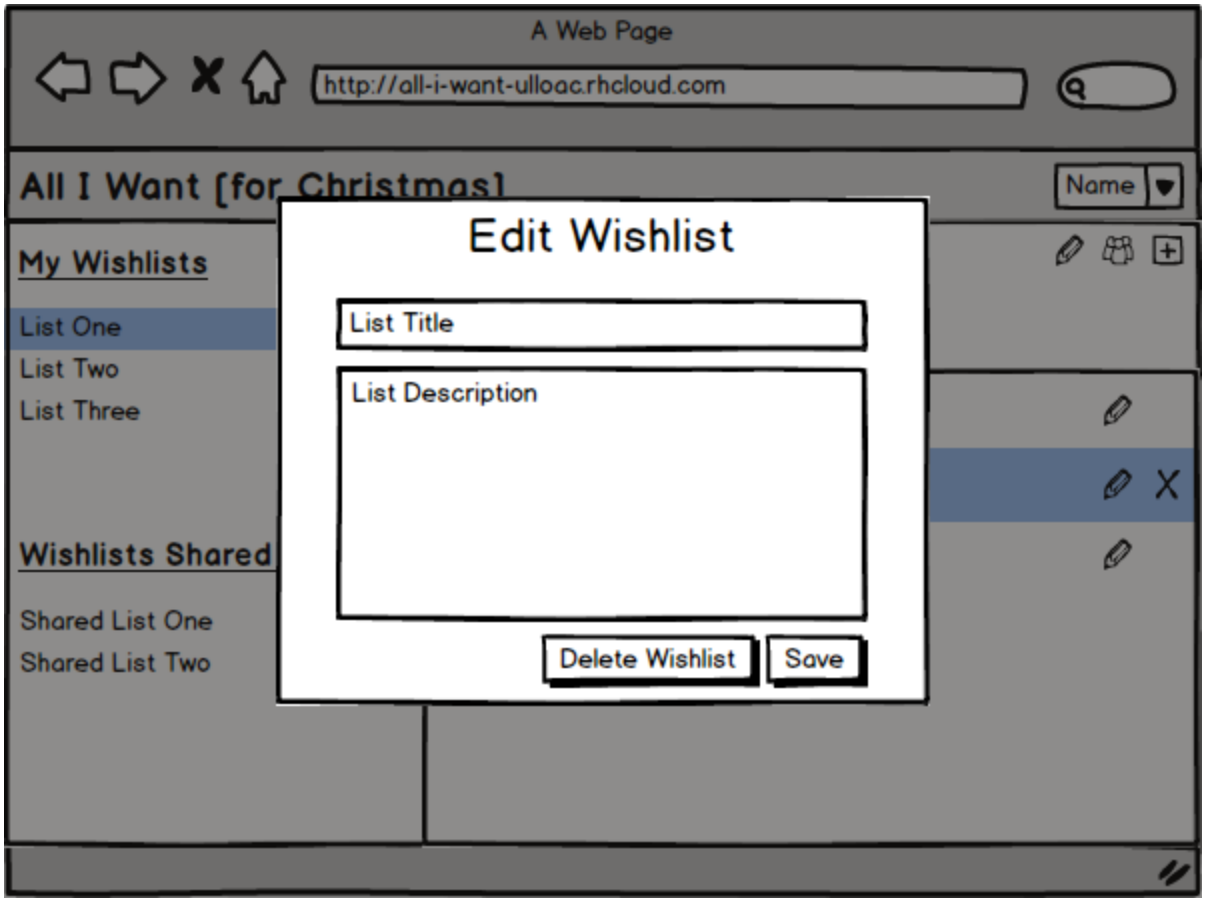
Create Wishlist Modal:



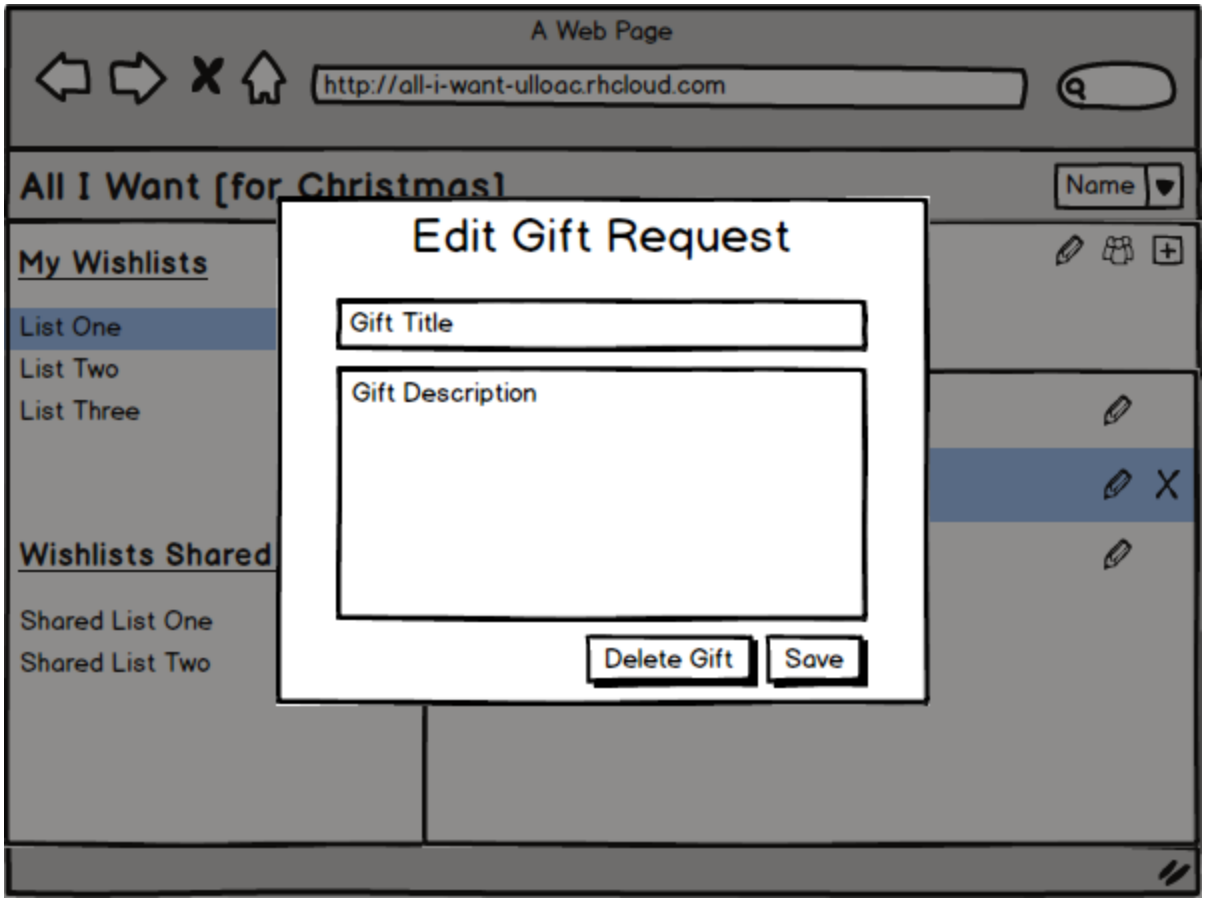
Add Gift to Wishlist Modal:



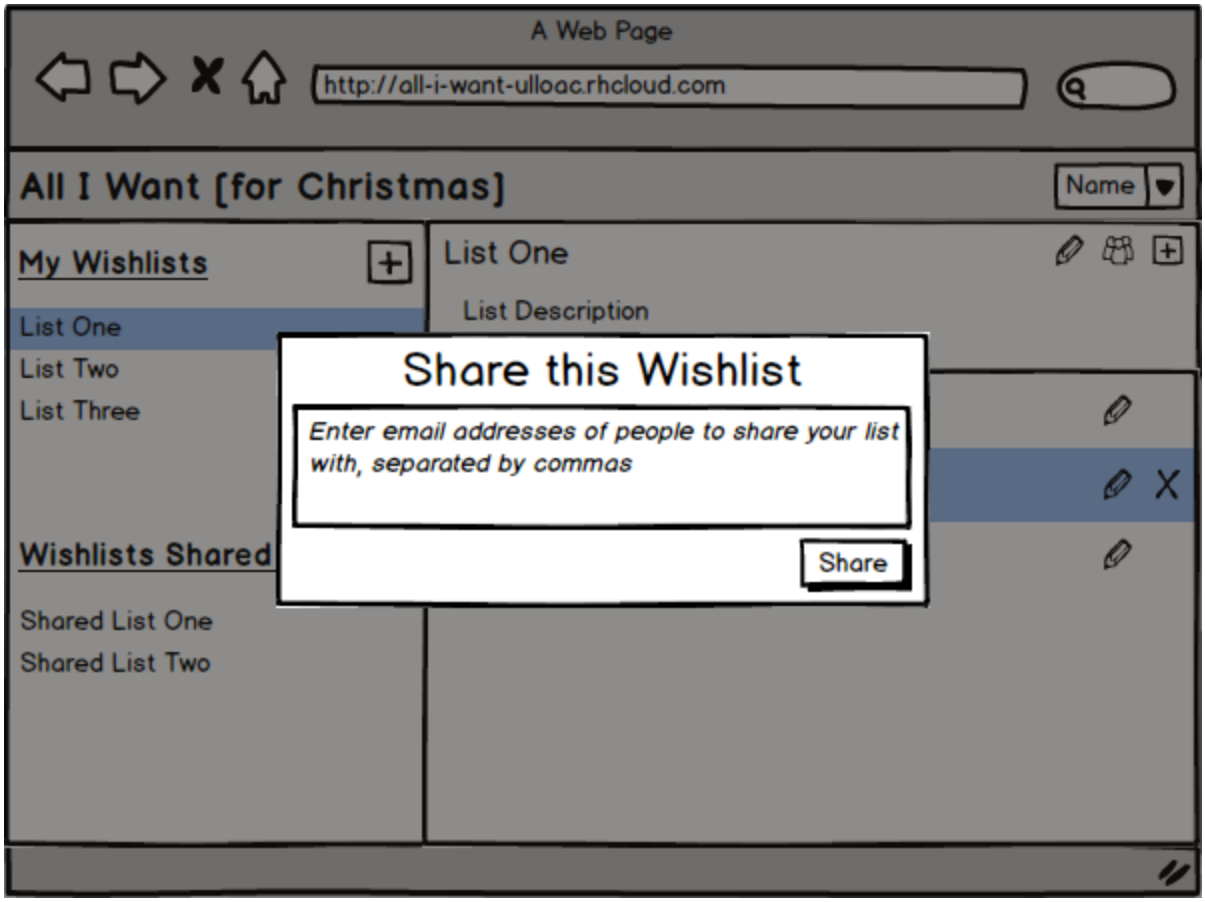
Edit Wishlist Modal:



Edit Gift Modal:

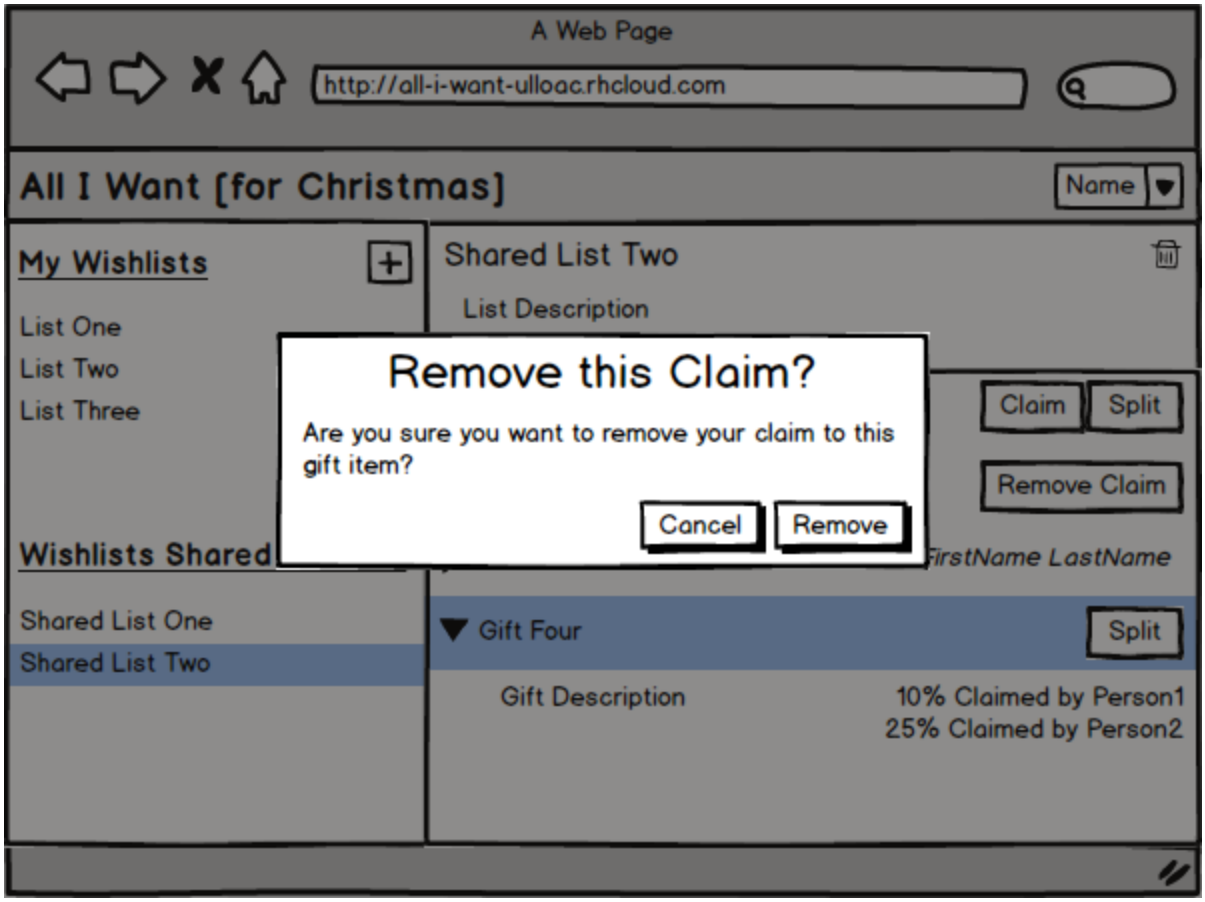


Share Wishlist Modal:

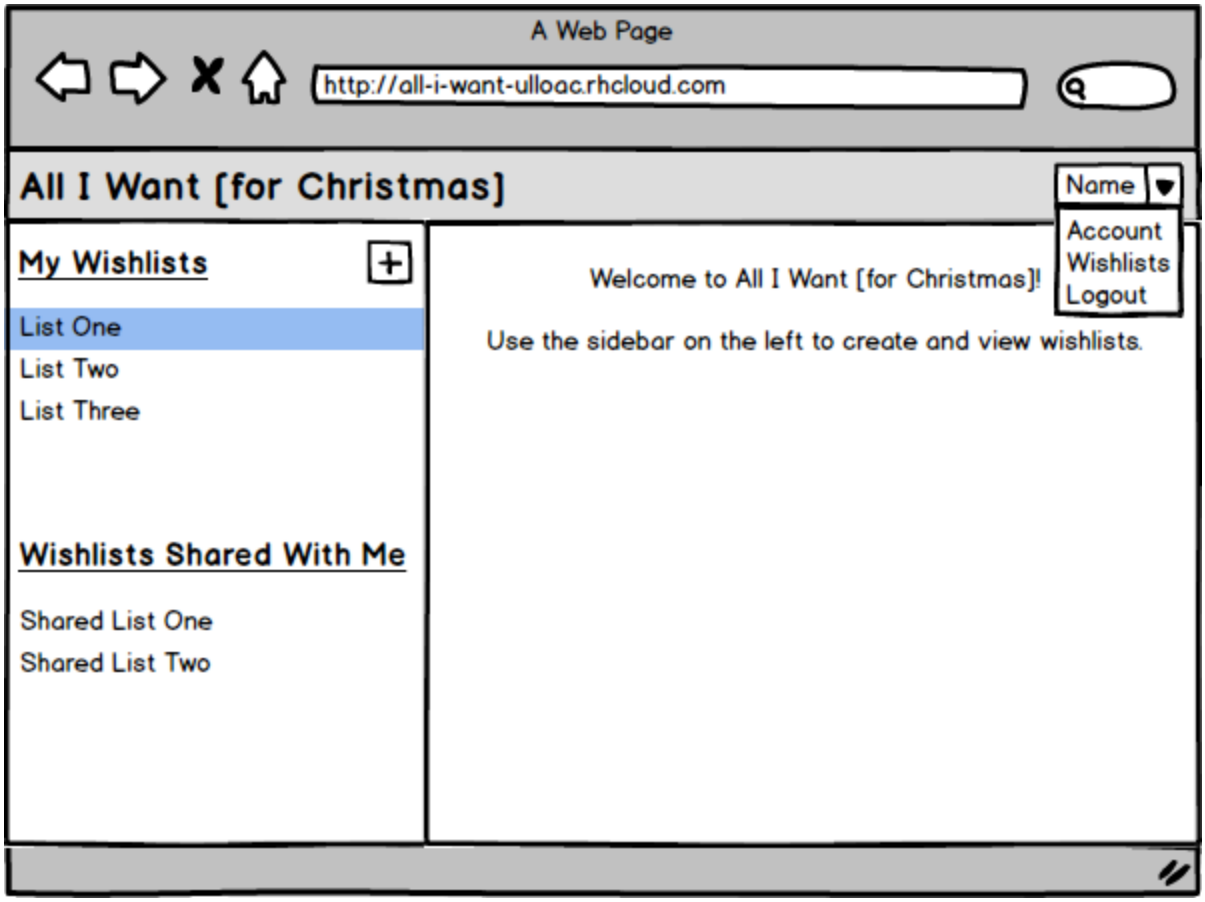


Split Claim Modal:

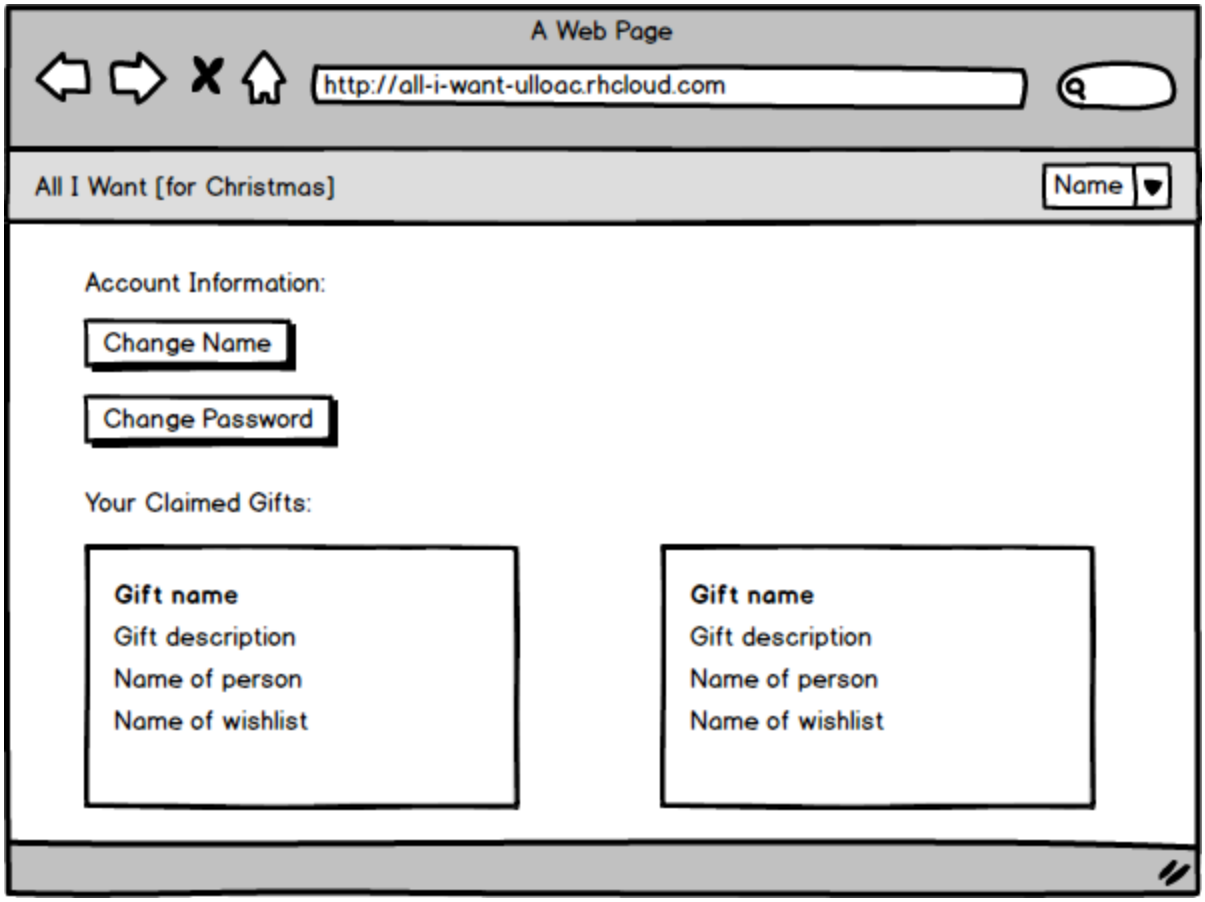
Remove Claim Confirmation Modal:



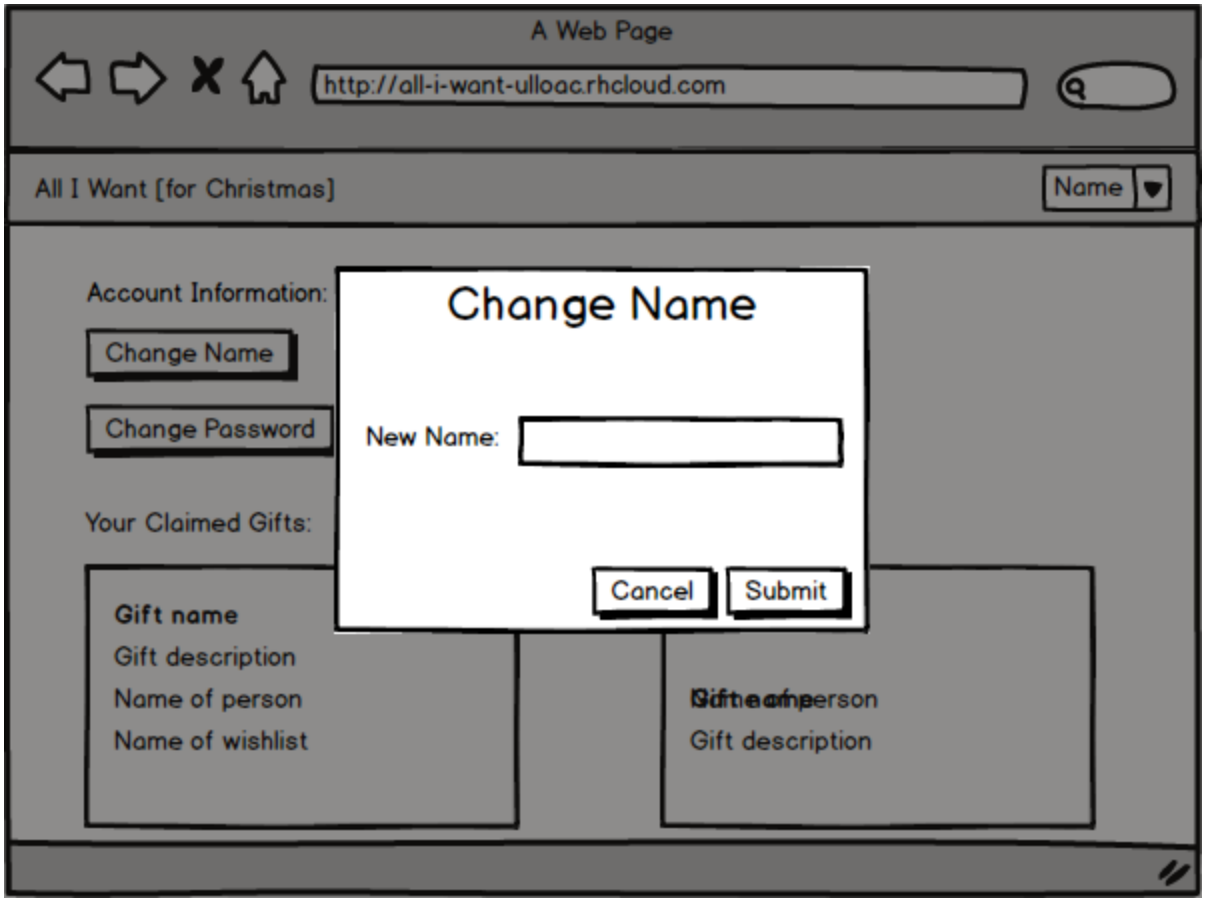
Open Navigation Dropdown:



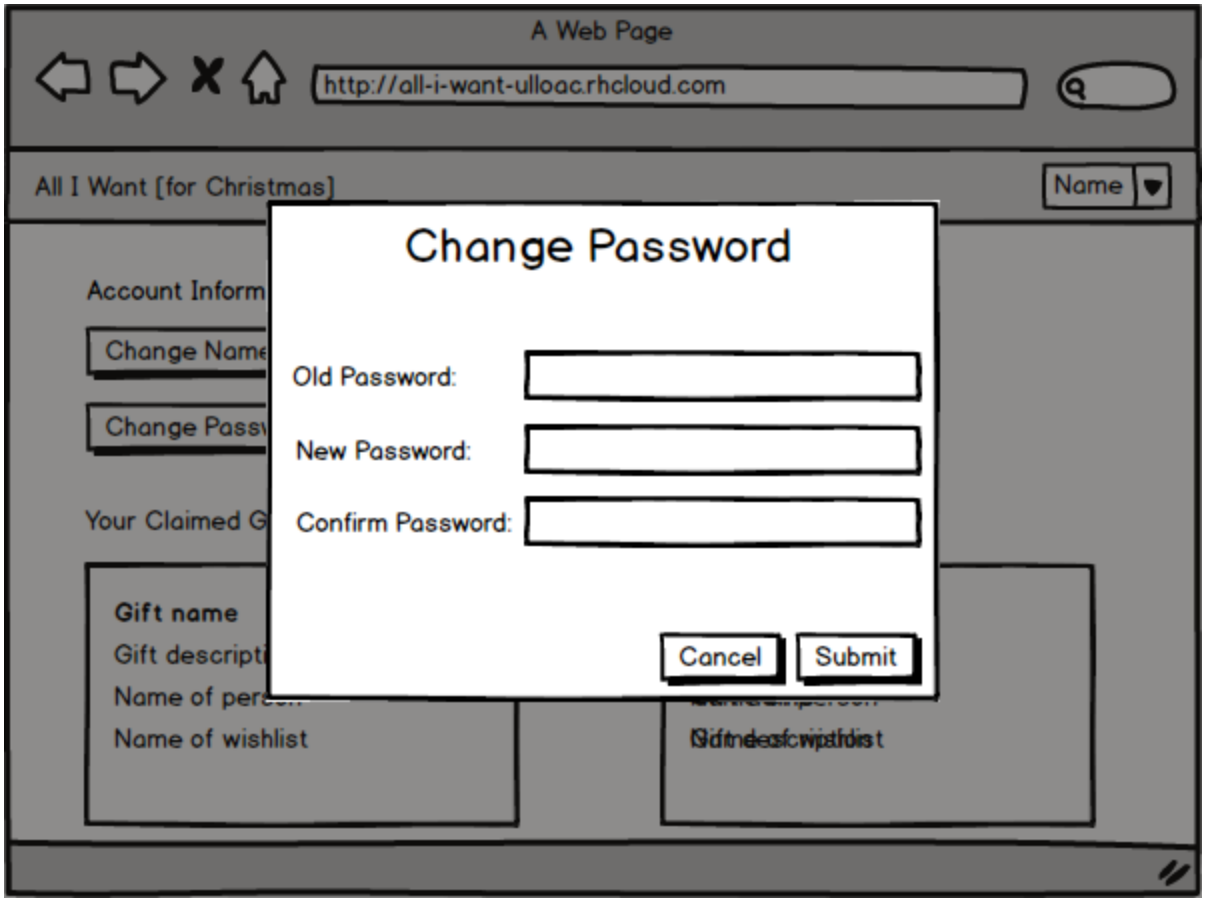
Account Info Page:



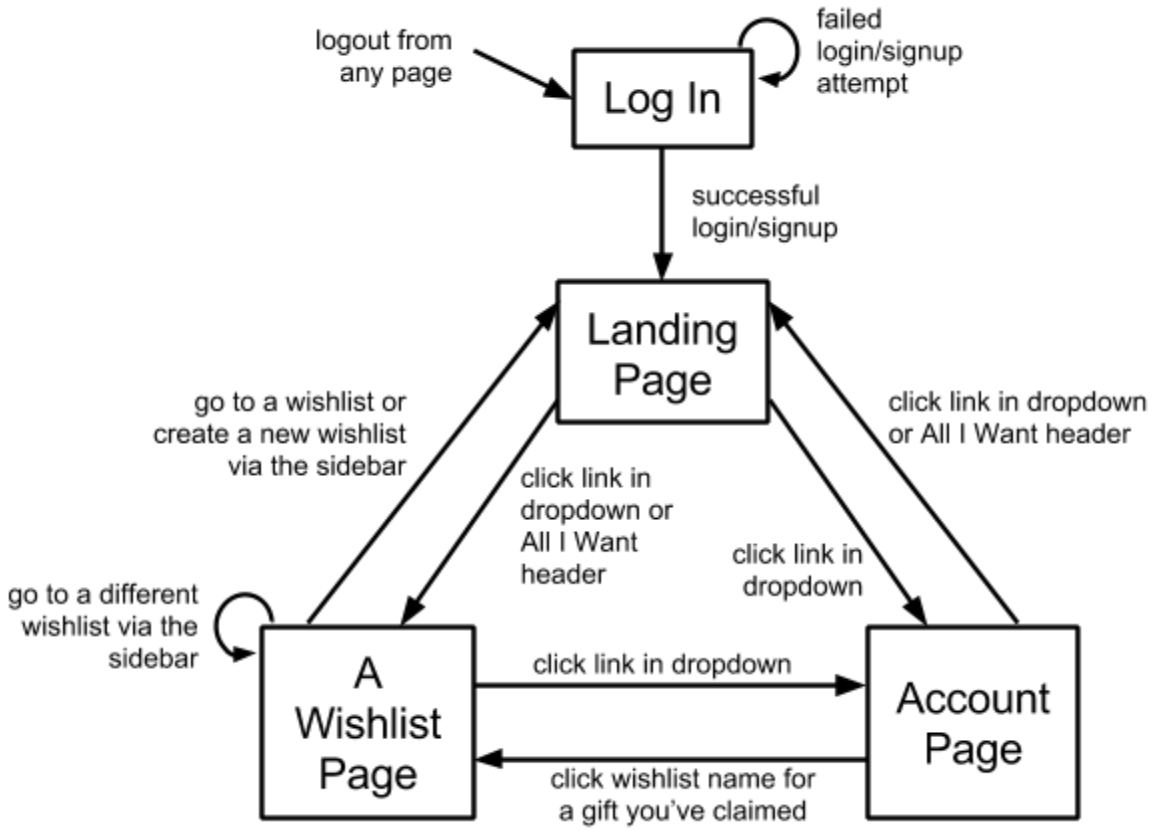
Edit User Name Modal:



Edit User Password Modal:



Page Flow Diagram



Design Challenges

Partially claimed gifts.

- If a gift is partially claimed, but a later user wants to claim the whole thing, should we allow them to override the split claim?
    - Prioritizing full claims makes sense to some degree, as it's ensuring the gift will be given (to the best of our ability)
    - However, everywhere else claims are ruled by first-come-first-served
    - Really, if someone has gotten there first and said they want to chip in to this, you shouldn't be able to take that away from them - but you can always contribute the rest
    - Solution: the second user can claim the remaining percentage or give a new unclaimed gift; we won't give override power to full claims over split claims
- How does a user select what percentage of a gift they want to claim?
    - Allowing for users to specify a percentage can lead to a strange unclaimed percentages (e.g. 33%+33%+33% is still not fully claimed)
    - We could just split the gift evenly amongst all contributors
        - but who decides how many contributors there should be?

        - and what if one user is willing to chip in more than another?
    - Solution: we will allow users to split a claim in 5% increments up to the remaining percentage
- Should users be able to unclaim split claims?
    - This could cause a previously 100% claimed gift to become unclaimed at the last minute and things would get messy
    - Once someone buys the gift knowing that the cost is 100% covered you shouldn't really be able to back out
    - But also we should allow users as much freedom as possible for changing their minds, since scenarios could arise in which this is necessary
    - Solution: users can unclaim a split gift until 100% of the gift has been claimed, at which point the user is locked in to their claim to avoid problems
- Should we allow split claims on public wishlists?
    - May introduce complications for the group actually arranging the purchase
    - But there are definitely use cases for this, for example a charity wanting to raise funds for a large purchase
    - Arranging the purchases is really outside of the scope and purposes of our site
    - Solution: split claims should still be allowed on public lists

Claiming Gifts

- Do we want to allow users to unclaim a gift after they've claimed it?
    - This is definitely a possible functionality that users could want in certain scenarios
    - But it's not great since other users who might have seen it claimed earlier might not come back to see it unclaimed, but there might not be anything we can do about this
    - Solution: users should be allowed to change their minds
- Should users have a list of gifts they have claimed? If so, when do they go away?
    - We don't want things you claimed like 3 years ago to still show up
    - A record could be kept by sending confirmations via email
    - We don't want to have to handle dates for wishlists because including a background scheduler for this isn't worth the added complexity
    - Solution: allow users to manually remove claimed gifts after giving them

Wishlist and Gift Info

- Should wishlists require address information for where to send gifts?
    - Will often be necessary information (for out of state relatives, convenience in shipping online orders, etc)
    - But users might not want to include their address - personal info
    - And it may not be strictly necessary - lists shared with local friends, lists for events, lists for local charities with drop-off boxes, etc.
    - Solution: wishlist description can contain address (and will suggest this as possibly important), but will not be required
- Should gifts include links to the products?

    - Would be convenient for things found online or that can be directly bought online
    - But might be inconvenient for wishlist creator to have to track down links for all of their gifts if they didn't have them initially
    - And we don't want to exclude gifts that had no specific products in mind (mystery novels, women's size medium sweaters, etc)
    - Solution: gift description can contain a link, but will not be required
- Should gifts include prices?
    - People might want to choose which gift to give based on a price range
    - But because of our decision to include gifts without specific product links, the user might have to estimate the cost or range themselves
    - Solution: we'll exclude prices on our site directly

Teamwork Plan

Stakeholders

The main stakeholders in this project are the potential users. We expect that the app will be most useful for families with children. Because we're not directly associating gifts with specific vendors, vendors are not notable stakeholders for this project.

Resources

The computational requirements for our project are not particularly intensive, and should be met by any modern computer. We will be developing and testing using Chrome as our primary browser.

There will be no direct cost to use or produce this app at our currently projected scale. Should the project's popularity explode, there might be costs incurred from needing a more scalable API for sending emails and from requiring more capable servers, which could be mitigated by selling advertising space.

The main time constraints for producing this app are of course the project deadlines. Besides that, several of our group members have other project classes requiring significant time investments during the next month, so we'll need to manage our commitments accordingly.

Tasks

Tasks for MVP:

- Deployment of app (probably on OpenShift) - little effort - Cristhian
- Creating Mongoose schemas - little effort - Cristhian
- Creating Log In page - moderate effort - Cristhian
    - Sessions framework (probably Passport)
    - UI (forms)
    - AJAX calls to sign up or log in
- Creating landing page/template - moderate effort - Dirk
    - dropdown linking to different pages
    - sidebar to access other wishlists (retrieving owned and shared lists from database)
    - text instructions and welcome
    - AJAX logout
- Creating Wishlist Page - substantial effort - Tricia and Kristin
    - displaying expandable list of gifts from database for the shown list
    - editing/deleting list if it's theirs

    - adding/editing/deleting gifts if it's theirs
    - sharing with other users if it's theirs (and adding it to lists those users can view in the database)
    - viewing and claiming gifts if it's not theirs (adding to their claimed gifts in the database)
    - creating new wishlists
- Creating Account Page - moderate effort - Dirk
    - display account info
    - display claimed gifts (from database) with links to associated lists
- Testing Framework - moderate effort - Cristhian
    - set up QUnit and basic tests
    - (everyone will write some specific tests)

Milestones:

- Tuesday, 11/11: Design Document due
- Friday, 11/14: Mongoose schemas and sessions (Log In page) completed
- Saturday, 11/15: Wishlist basic UI, wishlist and gift *creation* completed
- Sunday, 11/16: Landing page/template completed
- Monday, 11/17: Wishlist and account pages completed
- Tuesday, 11/18: Testing and deployment completed

Tasks Beyond MVP: [Will be allocated following completion and evaluation of the MVP]

- Adding public list feature
    - UI for option in wishlist creation, field in database
    - Any changes to how users interact with wishlists when they're public (TBD)
    - Optional page for discovering public lists within the app
- Adding split claims feature
    - implement adding/modifying/deleting split claims
    - UI for who has claimed what portion of a gift
    - email notifications to claiming users when 100% is reached
- Adding email notification feature
    - setup email framework
    - UI elements to trigger emails
    - Automatic notifications
- Modifying account info
    - UI for editing (modal)
    - AJAX calls to update database
- Unclaiming gifts
    - Add UI button
    - Database calls
- Security Mitigations
    - ensure all user input is cleaned before AJAX calls are executed
    - encrypt user passwords and emails within the database

    - implement email verification of passwords

Risks

- Concurrency of editing/claiming
    - Single document for each gift (including claims on it)
- Users finding out what gifts they're getting
    - Require unique email address for login
    - Wishlists can be shared with a certain set of users
- Removal of gifts someone has already claimed
    - Send the claimant a message about the change
- Authentication
    - The scope of authentication is larger than we've previously worked with, including database encryption and email authentication. We will find good APIs to facilitate this
- Email notifications
    - Finding an appropriate API that fulfills our requirements could present a risk.

Minimum Viable Product

Our minimum viable product will contain all of the necessary features to fulfill our app's purposes. It will be a complete online wishlist sharing product, missing only a few of the more implementation-heavy additional features. Because of this, our MVP may be more similar to existing solutions than our final product will be.

With our minimum viable product, users will be able to:

- sign up for an account
- log in or out
- create, modify, and delete private wishlists
- add gifts to, modify gifts on, and delete gifts from existing wishlists
- share wishlists with other users
- view wishlists that have been shared with them
- claim gifts from those wishlists
- view their account information
- view a list of gifts they've claimed

Our MVP will include all of our app's concepts except for Split Claims.

Issues postponed:

- no public lists
- no split claims
- no email notifications
- no editing account information

- no unclaiming of gifts
- security concerns? (we'll discuss with a TA if this is ok to postpone)