Use Case 1: Add Inventory Item(s)

Goal: User wants to add item to inventory
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to inventory page
Success end: Item is added to inventory
Failure end: Item is not added to inventory
Trigger: User opens inventory screen
Success Scenario:

1. User opens inventory screen
2. User chooses a main image for their item
3. User adds at least one tag to their item
4. User adds text description to their item
5. User selects ‘free’ or ‘for trade’ appropriately
6. User adds item to inventory
7. Item appears on their inventory screen

Extensions:

7a. Not all fields are filled properly
    i. ‘Add to inventory’ is unselectable

8a. Device has no connection to the remote server
    i. Device displays ‘could not connect to server’ error
    ii. App returns to add to inventory screen

Alternative Scenarios:

3a. User adds additional images to their item
7a. User cancels action
    i. User is returned to inventory screen

Use Case 2: Add Wishlist Item(s)

Goal: User wants to add item to wishlist
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to wishlist page
Success end: Item is added to wishlist
Failure end: Item is not added to wishlist
Trigger: User opens add to wishlist screen
Success Scenario:

1. User opens add to wishlist screen
2. User chooses a number of tags
3. User adds to wishlist
4. Tags appear on user's wishlist

Extensions:

5a. App cannot contact server

i. App displays 'cannot connect to server' error

ii. App returns to wishlist screen

Alternative Scenarios:

4a. User cancels

i. User returns to wishlist screen without adding items

Use Case 3: Browse by tag

Goal: User wants to browse items with given tag(s)
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to the home screen
Success end: Items corresponding to tag are displayed
Failure end: No items are displayed
Trigger: User opens browse screen
Success Scenario:

1. User opens browse screen
2. User selects a number of tags
3. User searches for selected tags
4. App contacts remote server, receives items matching selected tags
5. Items are displayed to user

Extensions:

5a. No items with matching tags found in the user's area

i. App displays 'No items found matching selected tags'

ii. App returns to browse screen.

Alternative Scenarios:

None.

Use Case 4: Browse matched users' inventories

Goal: User wants to browse the profile of a matched user
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to the home screen
Success end: The inventory of a matched user is displayed to the end user.
Failure end: No inventory displayed.
Trigger: User opens matches screen
Success Scenario:

1. User opens matches screen
2. App communicates with server, receives matched users
3. Matching profiles displayed to user
4. User selects a
5. Matched user's inventory is displayed to end user

Extensions:

2a. No matches found

i. App displays 'No matches yet - Try again later!'

Alternative Scenarios:

5a. User opens matched user's wishlist from their inventory screen

Use Case 5: Propose a trade from a user's inventory

Goal: User wants to trade with another user
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to another user's inventory
Success end: A new trade offer is created
Failure end: No new trade offer is created
Trigger: User opens 'offer trade' screen
Success Scenario:

1. User opens offer trade screen
2. The app presents a screen with both user's inventories visible
3. User adds items from each inventory to trade
4. User offers trade
5. Trade offer is sent to user 2

Extensions:

3a. One user has no valid items to add to trade

i. The 'offer trade' option is unavailable until at least one item has been added from each side (unless all requested or offered items are free)

Alternative Scenarios:

None.

Use Case 6: Trade with another user from the ‘browse’ screen

Goal: User wants to trade with a user from the ‘browse’ screen
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to the browse screen with items displayed
Success end: User initiates trade with other user
Failure end: Trade is not initiated
Trigger: User selects an item on the ‘browse’ screen
Success Scenario:

1. User selects item
2. Item description window opens
3. User opens inventory of other user
4. Inventory screen of other user is displayed
5. User opens offer trade screen
6. The app presents a screen with both user’s inventories visible
7. User adds items from each inventory to trade
8. User offers trade
9. Trade offer is sent to user 2

Extensions:

7a. One user has no valid items to add to trade

i. The ‘offer trade’ option is unavailable until at least one item has been added from each side (unless all requested or offered items are free)

Alternative Scenarios:

4a. User desires to view other user’s wishlist before trading

i. User accesses other user’s wishlist from their inventory screen
ii. User browses other user’s wishlsit
iii. User closes wishlist, returns to step 4

Use Case 7: Manage an existing trade offer

Goal: User wants to accept an existing trade offer
Primary Actor: Finders Keepers user
Scope: App
Level: User
Precondition: App is open to the home screen
Success end: User accepts a trade offer
Failure end: Trade offer cannot be accepted
Trigger: User opens offers screen
Success Scenario:

1. User opens offers screen
2. A list of pending outgoing and incoming offers is displayed
3. User selects an incoming offer
4. Offer details are displayed to user
5. User reviews details
6. User accepts
7. App provides contact details of other user, and sends push notification to other user that their offer has been accepted.

Extensions:

6a. Offer has been canceled
    i. App displays ‘offer canceled’
    ii. App returns to offers screen

Alternative Scenarios:

3a. User instead wishes to modify an outgoing offer
    i. User selects outgoing offer instead
    ii. User may now review and/or cancel outgoing offer
5a. User wants to review specific item descriptions
    i. User selects an item in offer details menu
    ii. Item description is displayed on screen
6a. User dislikes incoming offer
    i. User instead declines
    ii. Other user sent push notification that their offer was declined
    iii. User given option to reply with a counter-offer