DESIGN AND EXPERIENCE MATTERS -

1. Timeout thresholds - don't allow Alice to create her c2tx1 too late that could result in Bob spending Alice's but Bob's c1tx1 timing out not letting Alice spend it.

2. Don't allow Bob to spend Alice's c2tx1 too late because it could reveal his secret allowing Alice to spend Bob's c1tx1 but leave Bob's spend of Alice's c2tx1 too late due to congestion on ç2.

3. Tx creation sequence of Alice's client is a) spend of Bob's c1tx1, submit to server, then b) c2tx1. To prevent b) being created only in the event Alice's client fails before creating a).

4. UI - allow Bob to accept/reject the swap once Alice's c2tx1 is funded and ready to go.

5. UI - allow Alice to accept/reject the swap just before she creates her c2tx1.

6. Define the technical protocol for storing the password??? That we talked about on Friday night - when the initiator initiates the swap. (I forget the details - but was to do with using a public key to …. Aaaahg, I forgot!)

**REFUND Txs**

For Bob, we create a Tx1 then we create the first of two (redundant) refund Txs.

The first is created before Tx1 confirms - and it's sent to the server for the server to save it and broadcast it in the event that Tx1 times out and Bob wishes to redeem his funds. This way the server can check on Tx1 timeouts and automagically broadcast this first refund Tx.

If the Tx1 was maleated, then this first refund won't work and will require Bob to login and manually redeem his funds in Tx1 by NEED TO CHECK HOW WE SPEND TIMEOUTS - CAN THEY BE SET UP TO AUTO SPENDABLE BY BOB ONCE TIMEOUT OCCURS. IF THIS IS THE CASE, THEN NO FURTHER REFUND TXs ARE REQUIRED BECAUSE BOB ALREADY HAS ACCESS TO THESE FUNDS. AP TO EXPLORE.

ALICE Tx2

Alice should only be allowed to create her Tx2 that spends bob's Tx1 after Bob's TX1 is suitably confirmed. Reason being is that Alice's Tx2 refers to Bob's Tx1ID and if Bob's Tx1 is maleated, then the ID will be different and Alice's spend of Bob's Tx won't work.

UI FIXES REQUIRED -

1. Ok buttons. (Or “proceed” or something like that.) Currently the UI wireframes are very incomplete. See next point.
2. Better layout required on the UI - to reflect intuitive flow. (Though this may be fixed by the Shapeshift-like dialogue box flow we discussed Friday evening.) ####Kind of like frame3 and the way it is set out?

REQUIREMENTS -

1. Allow users to specify (coinmarketcap) published exchange rates, or select their own exchange rate. Allows users to come to their own agreement!

OTHER ACTIVITIES -

1. Write a FAQ. Define the topics for the FAQ.
2. Write a “how it works”
3. Write copy for UI - including instructions on how to use each UI.
4. Build and integrate the website headers, footers, menu etc. IKR!
5. Discover the best way to have dev, test and prod variants of the code in GitHub.
6. Write an “about us” page.