**Contract Design:**

The “tokenSwap.sol” contract is designed on the principle to conduct fair swap between 2 parties. While conducting a swap of tokens between 2 parties a couple of things could go wrong. For example, A approves to send tokens to B but B being dishonest and does not send his/her coins to A, ultimately taking A tokens without giving up his/her own, and some goes the other way around. So to cater this problem ”tokenSwap.sol'' is created so that the fair swap occurs. It will conduct the fair swap using the following mechanism:

* A will first approve ‘x’ amount of coins to “tokenSwap.sol” contract and B will first approve ‘y’ amount of coins to “tokenSwap.sol” contract
* Once this is done swap(x,y) function will called from the instance of “tokenSwap.sol” which
* The swap calls a function 2 times which basically does is that it sends A’s ‘x’ tokens to B and B’s ‘y’ tokens to A. The following function is used to for this manner, inside swap():
  + \_safeTransferFrom(token1, owner1, owner2, amount1); for A
  + \_safeTransferFrom(token2, owner2, owner1, amount2); for B

**Gas Estimate and Security Analysis:**

The gas consumed for “token.sol” and “tokenSwap.sol” is the following:

* token.sol Gas Consumed =
* tokenSwap.sol Gas Consumed =

The contract does the swap fairly and securely as nobody other than the 2 parties will have the tokens altered, whatsoever.