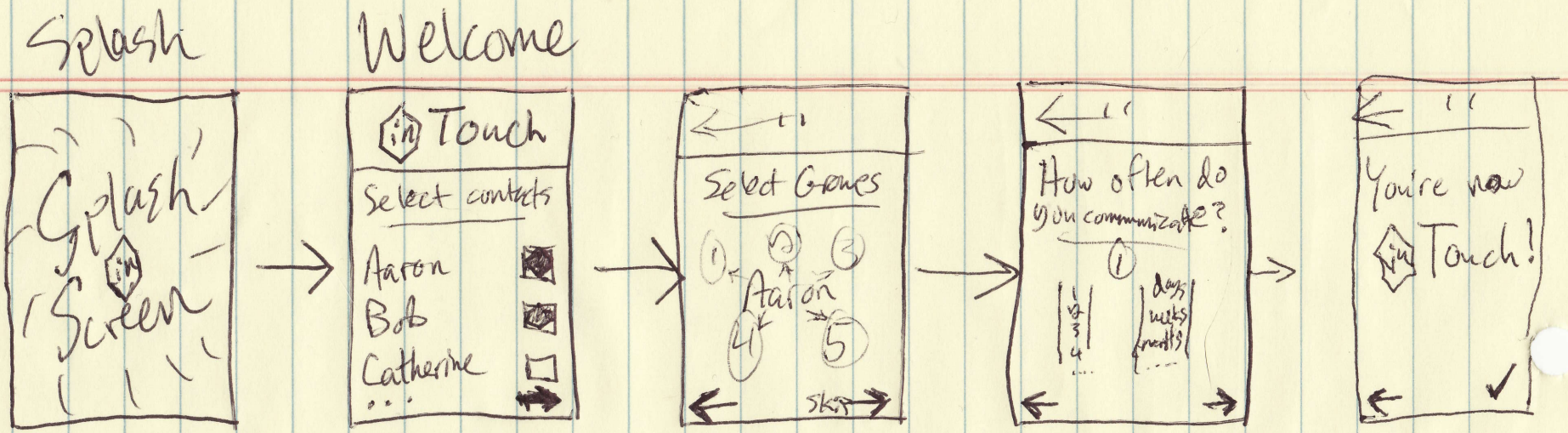

Splash
Splash Screen
Welcome
inTouch
Select contacts
Aaron
Bob
Catherine
Select Groups
Aaron
1
2
3
4
5
Skip
How often do you communicate?
1
1
2
3
4
days
weeks
months
You're now inTouch!

```
When time.hour=12:00
    Run Intouch()
    { lst_index = G.ArraySize
        for(ii=0; ii<lst_index;ii++)
          { for(jj=0; jj< G[ii].Arraysize; jj++)
            { count= G[ii][jj].object.count
              bounds= G[ii][jj].object.count
               Notify()
            }
          }
    }
```

```
Make_Node (input)
{ if check = true
    { Node = New Node
      Node.Name = Contact.Name
      Node.Number = Contact.Number
      Node.Group = input
      Make_Array_if_not_exist;
      if input != Null
      { G[in].ArraySize = G[in].ArraySize + 1 }
       switch (input)
         case 1
                Node.bound = val 1
                G[i] = Node
         case 2
                Node.bound = val 2
                :
         default
                bound = 30
    }
```

- Interact with call logs time & stamps

# run every day in background

Con_Array = new Array ⟹ dynamic maybe

SizeA = Size of Array

ii = 0

while (ii < size of Array )

{ count = A[ii].object.count

bounds = A[ii].object.bounds

~~function~~ Notify (a, )

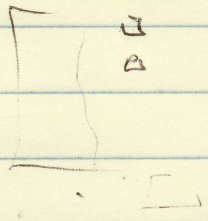
ii ++ } # repeat over and over

# Run when adding new contact

- Array size # if adding new contacts

if array size == array length

array length = array_length + round ( Array_length/3 )




# Actually setting up data structure

if check = true

Node = New Node ;

Node.Contact = Contact.Name

Node.Number = contact.number

~~Node~~

```
Utility (count, bound)
{ if count == Bound
    { pop-up window
      ignore = false
      answer = false
      if (ignored && ! Bound_reached)
        { count2t = count
          count = Bound/2
            Bound_reached = true }
      else if (ignored && Bound_reached)
        { count2t = count
          count = Bound/2 }
      if (answered)
      { pop-up window
          call_bool = false
          texts_bool = false
              . . .
        }
    }
}
```