Love at First Launch in a Bi Di App:

My biggest UX challenge and how I solved it.

What is Love at First Launch and Why does it Matter?

In this talk (link) at WWDC (year) engineer (name) talks about that critical moment when the user is so excited they are ready to take the leap.They jump to the App Store and hit the get button. That moment is where most apps fail.The percentage of Users who jump off the wagon is extremely high.

Why is that?

The answer is simple and backed up by study after study.

Putting the app developers interest above the users.
  1.asking for a login event before user enjoys the content- or
 2.asking for permission to track their location.
Those issues are relevant to all apps and it is good advice that we all should take note of.

My issue is not about **making demands** but about **creating confusion** or cognitive friction in your user's first experience. The way to understand the difference is this. If you invite someone on a date and ask them to pick you up and ask them to make the reservation that is making demands. However, If you make the reservation and you pick them up, but tell them you will pick them up at "zero hour plus twenty" that's cognitive friction. I have learned through experience that you can do everything right and the user can love the experience but they

can get stuck inside a scene where they simply don't know how to proceed. The reason this can happen In a BI Di app (Bi Lingual Bi Directional) is that sometimes the user forgets that they are in a RTL situation and instead try to gesture in a LTR manner. They then come to the conclusion that the app is broken. I encountered this phenomenon early and often in my Daven4me app.
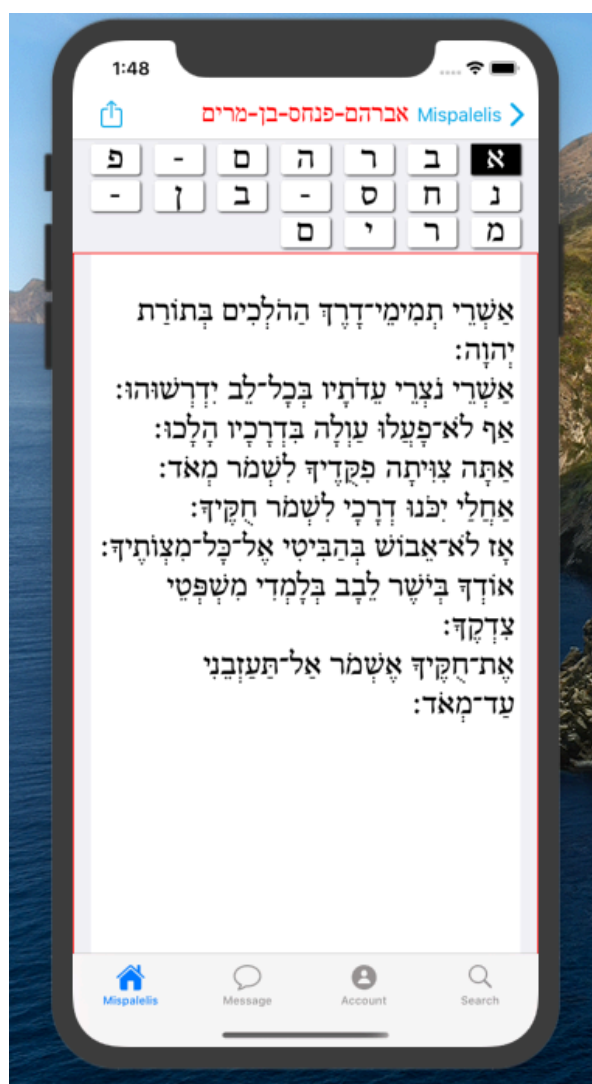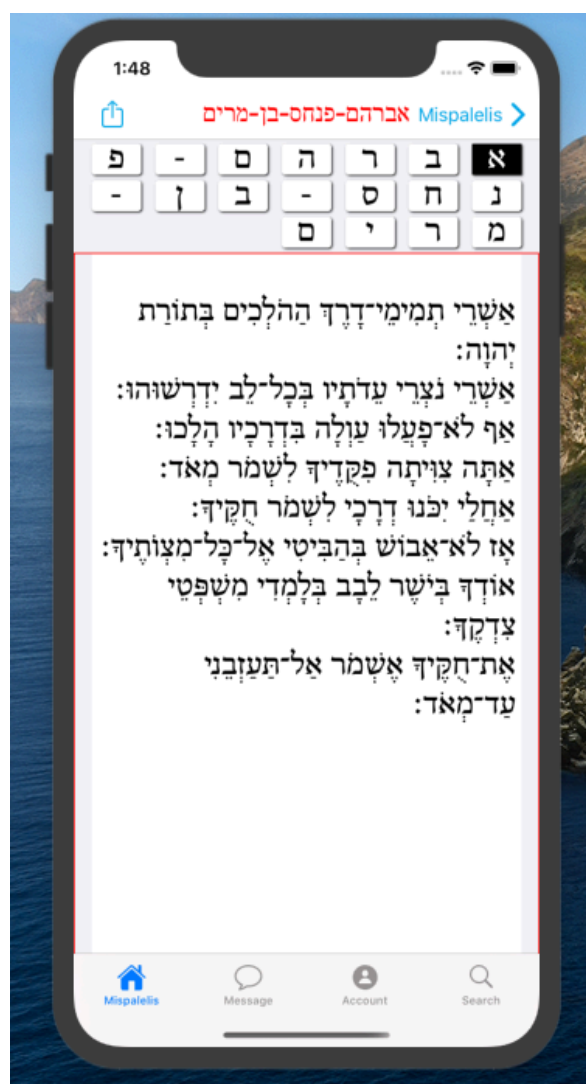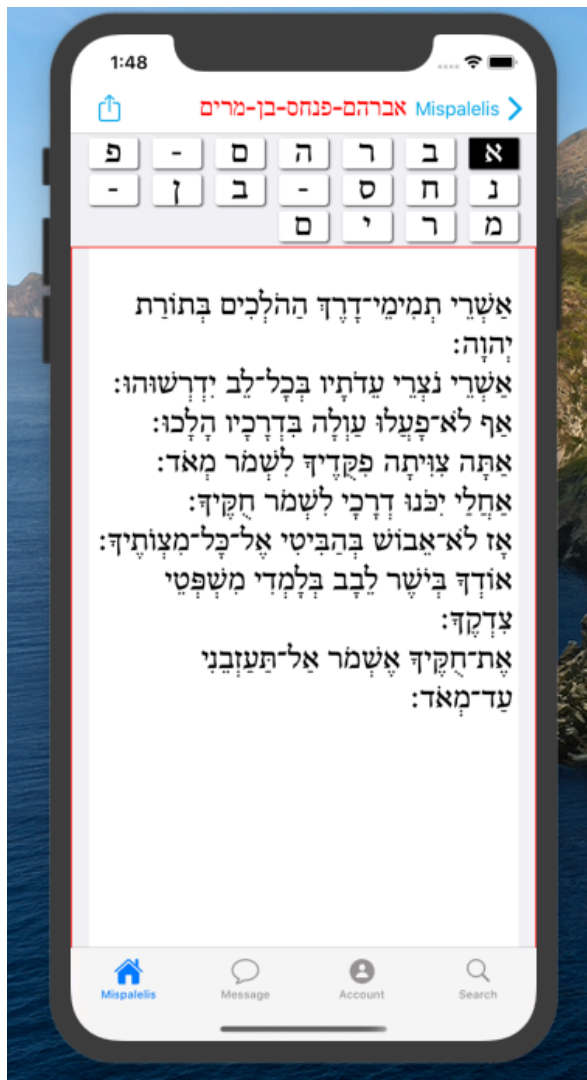

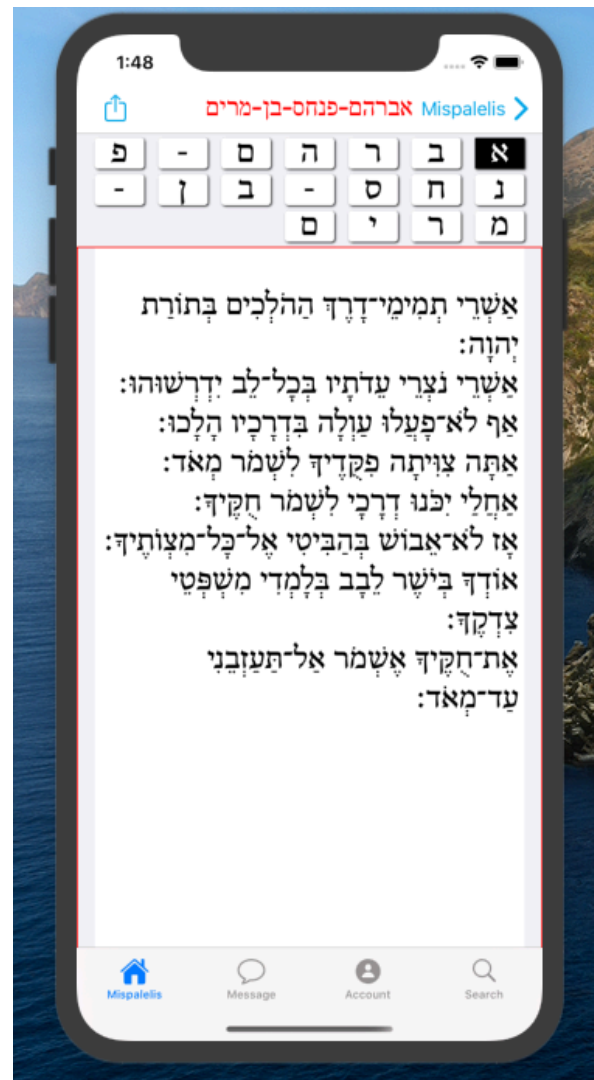**RIGHT TO LEFT SCROLL**   **RIGHT TO LEFT SCROLL**

I tried to provide a custom paging experience for them to scroll horizontally through the pages of a book.

(Screen Shots)
My users are very familiar with that metaphor because their home screen works that way. Also the Apple Books app works that way. However, their home screen and their books app starts at the left

RIGHT TO LEFT SCROLL

RIGHT TO LEFT SCROLL

and you swipe left to proceed. (Pictures)

In this scene you start on the right edge and swipe right to proceed (if all goes according fo plan:)

        The way this came to my attention was when I was pursuing opportunities to promote my app to a larger audience through influencers in my target community. They asked me to simply send them a link so they could experience the first launch themselves without me
guiding them.

Merril F:

"The bottom screen Is stuck..I think it's broken"

Tamar P:

"It's really great I'm on the detail scene and I'm .... Im ... it's not moving… Oh... I was swiping to the left cause that's what I'm used to doing "

Paysach K:

"I just don't know what you want me to do on this screen..how do I get to the next page?"
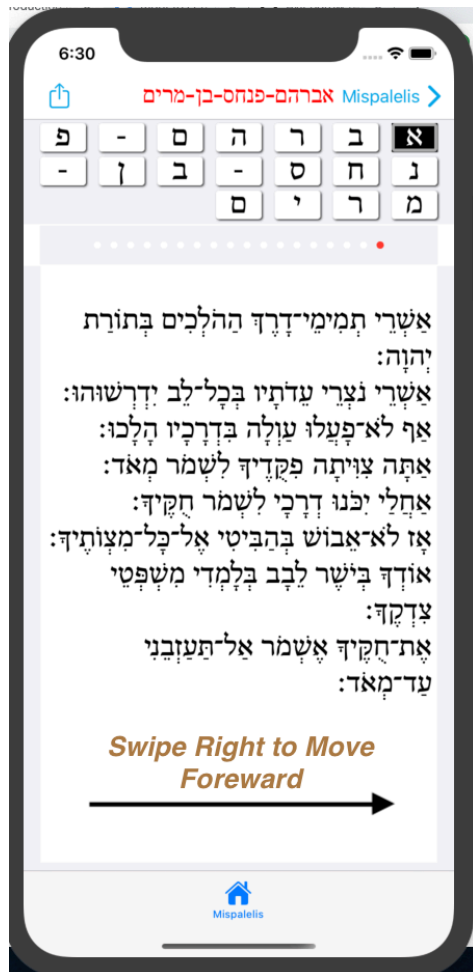
So my UX challenge was:

how can I hint to the user that
a- this content is **pageable**-
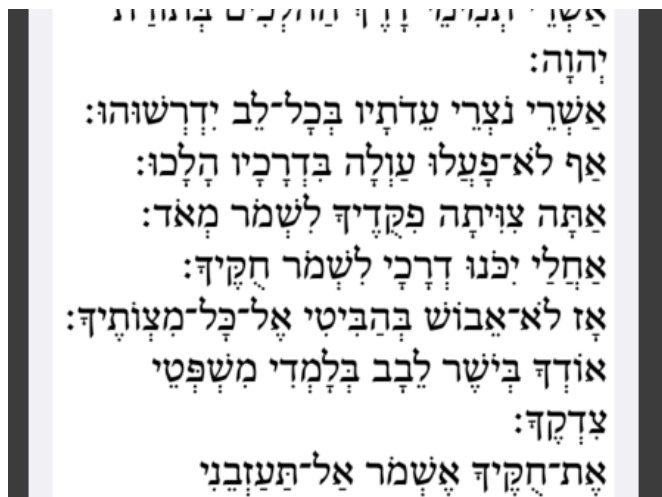b- that it is **horizontally** pageAble - and
C - that you are all the way on the right and need to **drag right to left proceed** and drag left **to** right to back up.

I wanted to help my user but I didn't want to over do it. My brute force approach was this arrow

but I still wasn't positive the user would digest it. I just don't have a lot of faith in people reading instruction in order to learn how to operate an app. I've given a lot of speeches in my career and I don't thing speeches is where people learn. When I want people to learn I give them an actionable assignment and guide them through the work until they can do it on their own. Talking is not teaching. So if talking isn't teaching then for sure popOvers with text are not the best way to make sure users are learning. So a popover with a text bubble wasn't the answer.

My fancy brute force approach with a high probability of impact was apple's animating swipe gradient (picture)

יְהֹוָה:

אַשְׁרֵי נֹצְרֵי עֵדֹתָיו בְּכָל־לֵב יִדְרְשׁוּהוּ:

אַף לֹא־פָעֲלוּ עַוְלָה בִּדְרָכָיו הָלָכוּ:

אַתָּה צִוִּיתָה פִקֻּדֶיךָ לִשְׁמֹר מְאֹד:

אַחֲלַי יִכֹּנוּ דְרָכָי לִשְׁמֹר חֻקֶּיךָ:

אָז לֹא־אֵבוֹשׁ בְּהַבִּיטִי אֶל־כָּל־מִצְוֹתֶיךָ:

אוֹדְךָ בְּיֹשֶׁר לֵבָב בְּלָמְדִי מִשְׁפְּטֵי צִדְקֶךָ:

אֶת־חֻקֶּיךָ אֶשְׁמֹר אַל־תַּעַזְבֵנִי

TYPE TO ENTER A CAPTION.

That definitely would work because I know it worked for Apple on millions of users …. including me, myself, and I. I decided to implement it for the sake of my portfolio since its pretty darn cool but for the real world I wanted something more subtle. I went back to my favorite question "How did apple do this?". Looking back to the design section in the WWDC app I saw this WWDC (date( ) session on buttons and visual cues by (name) … and I knew that I had found the elegant way forward.
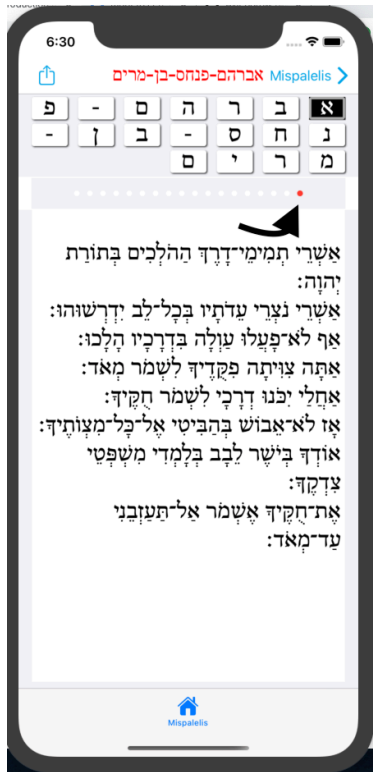
He asked
What is a page control?

It's a tell to the user "this is pageable content"

Great:
I wanted to tell my user "this is pageable..." so
Page control it would be.

But how do I "cue" that we need to page by swiping right?

So using the page control as my educational tool I gave it a bit of a pop (picture)


But cognitive friction is a dangerous thing if you want love at first launch so I decided to give them one last visual cue.

In fact, I wanted something better than a visual cue. I wanted them to actually take an action that would educate them in how this horizontal paging mechanism was intended to work.
I had learned from my years teaching 8th grade that when I'm talking they aren't learning. When I stop talking and they start doing…. they start learning. Sooooo I deliberately slid the scrollView just a bit offCenter to force the user to drag it back into proper place.

As a teacher I sometimes will ask isn't it a fact that 2 + 2 = 5.5?

Then the class says "no its 4 teach!" Good I now know you mastered that lesson because you took an action that confirmed it.
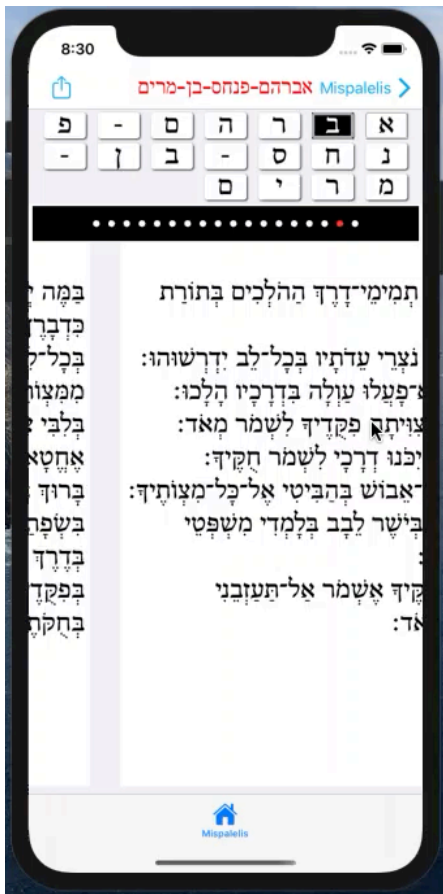
At this point the problem was solved but I had a nagging feeling that I had one last piece of UI magic to present. I like to model my UIs after popular Apple UI s like the Calendar app. Users expect all apps to behave like their Apple Apps that come with their device. So to meet that standard I set off to.....Activate the top collection view grid. So My final coup de grace was to give the user a completely different way to "get unstuck" by tapping the top control to scroll through the pages of the lower view controller. (Picture)
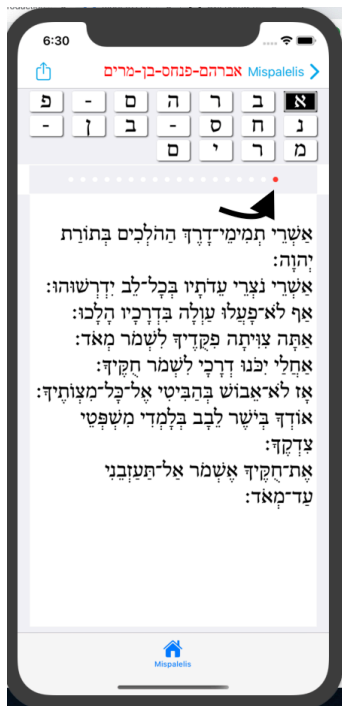
My biggest takeaway is that in Bi Di User interface you need to constantly make sure that your users know when they are in an RTL mode and when they are in LTR territory. The user will catch on fast but the very first launch is where 90 % of the danger lies. Test it by having them download the app and use it without any coaching. When you identify a pain point or a friction point look at how Apple designers solved that problem in their famous apps that users are already familiar with. Most importantly, remember that ***you are the teacher and the user is a student*** who is just trying to learn how to enjoy your awesome creation.

In conclusion let me leave you with the line from the theme song to Titanic.

"Love can touch us **one time** and **last for a lifetime**"-

So make that "first touch" be a touch of pure love:)