



Welcome to Calgary!

DESIGNING A KIOSK FOR NEW IMMIGRANTS

Daniel Armstrong (djarm2009@gmail.com), Debbie Macrohon
(dtmacroh@ucalgary.ca), Joanna Huynh (jhu.mai18@gmail.com)

Human-Computer Interaction I | April 9, 2016

Table of Contents

Visual Wireframing Prototype.....	3
Redesign Rationale	3
Lightweight User feedback	4
‘Horizontal’ Prototype.....	5
Design rationale.....	5
User Evaluation of ‘Horizontal’ Prototype	6
Justification of Evaluation Methodology.....	6
Evaluation Protocol.....	6
Evaluation Findings	7
Vertical Prototype	9
Final Design Rationale.....	9
Self-assessment of Quality	11
Heuristic Evaluation Protocol.....	12
Results of Heuristic Evaluation.....	13
Looking Forward.....	16
Appendix	17
Portfolio 1 Contents	17
Wireframe Prototype Screenshots	18
‘Horizontal’ Prototype Screenshots.....	20
‘Vertical’ Prototype Screenshots	22

Visual Wireframing Prototype

REDESIGN RATIONALE

The scenario we were trying to address was that an immigrant user would pass by the kiosk and would be interested in one of the categories mentioned in our kiosk. One scenario that we had was that an immigrant wants to learn more about Calgary so he would search through information. If the user wants to participate or experience an activity to further understand the concept, they could search through places of interests in the guide or they could find events that are relevant. The main issue that arose from our paper prototypes was that they were too specialized so not only was the information provided, but we also accomplished in-depth interactions that would best be accomplished using an external program. For example, we had a prototype that would be focused around transit; this prototype would be able to use the user's location to provide simple suggestions on transit routes to reach their destination. Implementing this prototype would mean that we were trying to create a transit kiosk, where our focus should instead be providing information for immigrants.

We also removed the Cost Management sections because the information we could find didn't fit into the scenario which we envisioned (a walk-up kiosk). In addition, we were unsure whether the information that would be contained in this section would be meaningful and useful for the user.

One issue we encountered was deciding on the layout of the prototype. We had to create a kiosk, but our paper prototypes had various layouts, such as a website layout and mobile application layout. We had decided on a mobile application layout that was wide enough to display larger information blocks, with similar size to a tablet display. Since we agreed on a mobile application layout, we used a button panel to allow navigations, rather than a sidebar using keywords. Because our paper prototypes were covering such a variety of topics, we combined the promising features of all of them to obtain the design we currently have as a wireframe prototype.

LIGHTWEIGHT USER FEEDBACK

For the lightweight user feedback that we received from testing our wireframe on users, we found that our testers could easily navigate through guides, info and events primarily because they were simple and straightforward. A minor issue was that the user would have to continuously press the back button until they would return to the main home page; this is not ideal, since a user would want easy access to a frequently visited page. This could be solved by adding a home button so that the main page would be one click instead of multiple. The user would also attempt to press the buttons containing categories that were not yet implemented; a solution to this problem would be to grey out the selection so that we could inform the user that it was currently inaccessible.

For our next iteration, we need to improve the design of our prototype so that we can reduce the amount of clicks to navigate through the interface. We need to add a home button so that the user can return to the home page quickly, and possibly add shortcuts to the main three focuses of our prototype on a toolbar so that the user can access these areas just as quickly. The events and the selection bar could be combined into one page so that they user wouldn't have to keep pressing the back button to change a selection. We need to improve the quality of the information presented in the prototype. The aesthetic of the prototype could be improved such that it will contain more colour, and we could definitely organize elements in a more aesthetically-pleasing way. We could also add photos to correspond with the information presented, and present descriptions to aid with the understanding of context. We definitely need to try and implement each category such that they will all contain at least one entry so that all the buttons will have at least some functionality.

‘Horizontal’ Prototype

DESIGN RATIONALE

For the horizontal prototype, we focused on light implementation on all the features. The main problem was our scope was too large and integrated too much functionality, and several key problems we attempted to tackle would have been better resolved by a separate stand alone application.

Our application includes four discrete categories of information: Places of Interest, About Calgary, Socializing and Events and Useful Links. Stemming from the wireframe, we renamed Calgary Guide into Places of Interest. We also renamed Calgary Info to Info about Calgary, and Events into Socializing and Events. We renamed all these categories to give users a better idea of what kinds of information is contained in each category, to minimize confusion and ambiguity.

Some other changes contrasting to the wireframe include a Useful Links page, removing transit information and cost management. The Useful Links page includes some sources that a first time immigrant might possibly need, such as the Calgary Transit website and Immigration Services. We removed cost management because it was on the lower end of the priority scale for potential users that we have interviewed, and led to divergent ideas in terms of how the user might approach the application. For example, if someone is using a mall kiosk, they would find locations of various stores, washrooms and places to eat. However, chances are that they will not expect or search for cost management tips. A potential user seeing our kiosk for the first time should walk up to it expecting to find general information about Calgary, and that is what we attempted to show in this version of our application.

User Evaluation of 'Horizontal' Prototype

JUSTIFICATION OF EVALUATION METHODOLOGY

Our evaluation methodology evolved naturally from our knowledge of how a usability evaluation is conducted, as given in class. We searched first for a piece of software that had a similar set of features as our horizontal prototype; this would end up being used as a control mechanism. Our choice of control ended up being a New York City map and travel guide, NYCGo. (<http://www.nycgo.com/>) If we had to re-do this testing, we would likely have searched harder for a similar public map for Calgary, for extra compatibility. We then went over our prototype's feature listings, and used them to generate a sequence of tasks for each version. The primary intent behind a number of these tasks was to prompt comments, gripes, musings, and other observations. They would also give us a set of potentially-useful metrics for how we can improve the design of our project. If there are any specific elements of the user interface that cause hang-ups or missed/mistaken inputs, we can see these as they happen, and fix them for future versions. Finally, they would also let us see the meandering path of how a user manipulates the system, and consider how to lay out the collection of data we have not yet displayed. This is most useful in considering how we should deal with features that hadn't been implemented at time of evaluation.

EVALUATION PROTOCOL

The procedure we used was a series of five tasks per program, plus a couple of questions at the end of the tests to prompt additional information from the user (if any). They were, in order:

1. Explore the given interface for 1 minute
2. Return to the home page of the interface
3. Search through pages in order to find an event within an area (for our control, it was in the Brooklyn area, while the prototype had Forest Lawn as the location)
4. Find some cultural food to eat (with the aim of going to the page listing various restaurants)
5. Find more information on the city's public transport system

After these questions had been asked, first for the NYC site and later for our prototype, we would close the interfaces and ask three questions:

What were the labels along the top bar of both programs?

Which were your favourite and least favourite aspects of both programs? and

Do you have any final comments about how you used these programs?

Our tests were run with two people for each test. One person would administer the testing and guide the testee through each of the tasks, while the other captured a video recording of the test on their phone or tablet, sending these videos to a sharing site for later analysis. The tests were done on a laptop with touch-screen capability, to simulate program usage on an actual kiosk.

We did a total of five tests, with participants being representatives from our original user research group. We graded each task separately, with our two metrics being ‘time taken to complete a task’ and ‘number of clicks (touches) used in completing the task’. Then, after all measurements had been taken, we calculated a third measure intended for determining page simplicity: clicks per second for a task. We also attempted to use a ‘number of page visits during a task’ metric, but this quickly became extremely lopsided and would not give usable data for analysis.

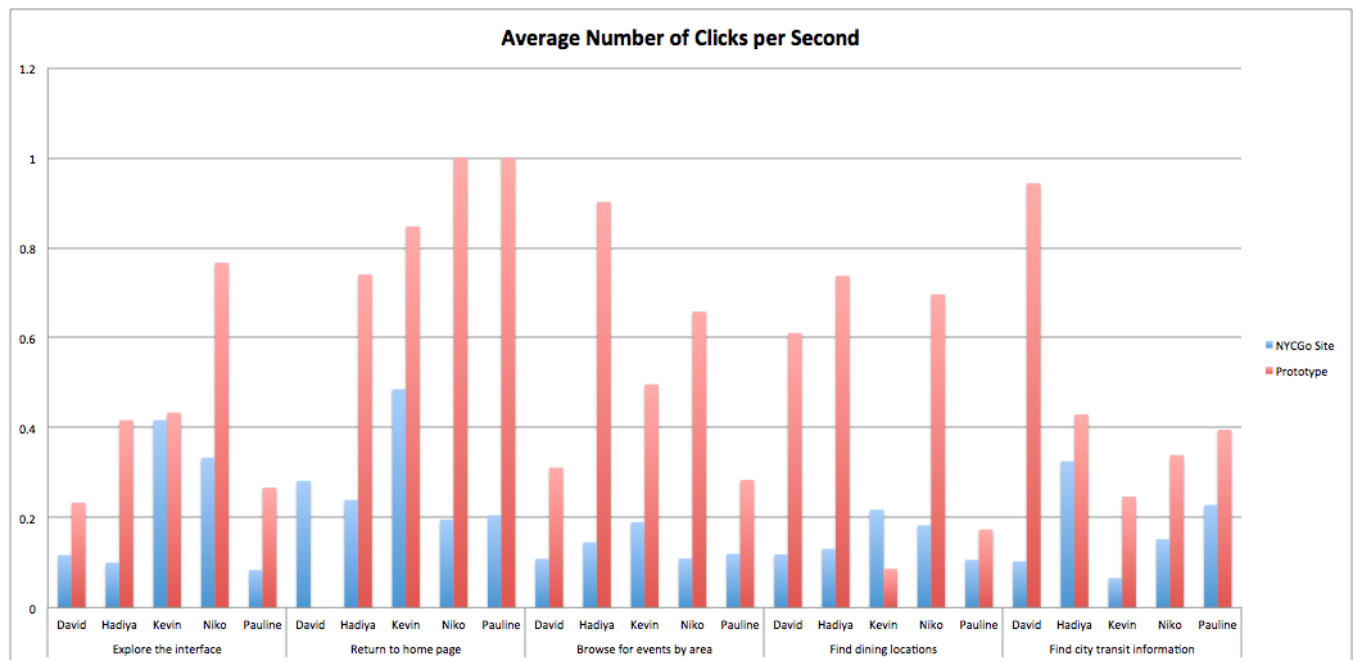
EVALUATION FINDINGS

The most agreed-upon asset of our prototype’s presentation is that its layout was extremely simple: a main page with four buttons, and each of the pages accessed through those buttons had easily-identifiable interactions.

On the other hand, half our users agreed that our choices in naming the main pages made it hard to determine where some sets of information would be located. This can be seen in our fifth test, where we tell people to search for transit information; if users had not already seen the option within the Useful Links page, they would start their search within each page of Info about Calgary. They also provided a number of interface improvement suggestions; our prototype was very colour-bland and sterile, and the information was presented with a rather plain layout. They also advised adding some sort of decoration to the landing page; it had

previously been the source of the main directory buttons, but was made redundant with the addition of our header bar. The third suggestion given was to consider switching to a different location selection method; on our events page, it was using a drop-down list of neighbourhoods to choose from, which a couple of testers considered potentially confusing for someone unfamiliar with the area.

As far as numerical analysis is concerned, this set of graphs best demonstrates the difference in speed and accessibility between the two software sets:



In this set of graphs, we display the amount of clicks per second each user took on each task.

As is visible, the prototype has (on average) a higher count of clicks per second. We can interpret this in several ways:

- The interface has a high number of interactable elements, requiring frequent inputs to view data
- The interface is devoid of content, and the simple pages can be read more quickly than can be expected in a content-complete site
- Pages within the interface are easily recognizable, leading to mistaken inputs being corrected in very short order
- Users have viewed the relevant information before, thus minimizing the amount of time it takes to find it again

- Users can intuit more easily the location of data based on classifications, reaching it faster.

We are inclined to believe, from the qualitative feedback and comments we received, that the factors which most contributed to these results were items 2 and 4. Only one tester ended up rapidly pressing buttons in the hope that he'd find the solution, which proved somewhat ineffective.

Vertical Prototype

FINAL DESIGN RATIONALE

From the horizontal prototype, we got the following feedback: "What do you want the users to take away after using the kiosk?" In order to follow this feedback, we decided to focus our vertical prototype on the Socializing and Events page. Our first attempt at scoping out this element was the creation of more filters: cost range, group size, participant age, participatory vs observation focus, type of event, etc. However, talking with users of similar systems revealed that providing too many filter options would likely lead to confusion at the system complexity, and result in a less-informed user. So, we chose to limit our list to four frequent and simple-to-understand filters: Area, Age, Culture, and Display Free Events.

We focused on the category Socializing and Events because our users' primary concern was to be more integrated into society. During their first few years here in Calgary, they felt lonely and wished they were more involved in Calgary culture, or met people with the same culture and interests as them. Having this need met would make them less homesick and lonely. As our project is focused on human-centered design, and not on Event Planning, the best way we can accommodate these users is by informing them about existing events that match their needs/requests.

Focusing on Socializing and Events, we changed our area filter from a drop down list to a map of Calgary (using regions rather than communities as selection criteria), following the

feedback we got from our horizontal prototype. In that prototype, searching by area would entail looking through a drop down menu of all the communities that we have in Calgary. We received feedback saying that it would be nicer if we had a visual representation of where the actual communities are. This would not be possible on the scale requested, as the quantity of information required to make a community map was not plausible within given time constraints. Instead, we limited the area selection to another frequently-used criteria: quadrants of the city. Clicking on the Area Selection filter button will now bring up a visual representation of the city, divided into the four quadrants. Selecting one of these regions would result in that region being highlighted, letting the user know which quadrant they selected before they confirm the search under that filter.

In addition we added a free filter and age filter to add more variety in searching for events. In comparing the kinds of information we had to that provided by NYCGo (www.nycgo.com, our control website in our usability evaluation), we discovered that our control website had filters for free events, and has the ability to filter by age as well. In revisiting the conversations we had with our potential users in the preliminary interviews, we thought that integrating a free events filter in our events page would be helpful for immigrants who want to keep their costs low. We included the age filter for immigrants who want events that cater to a specific age group. After all, events catering towards children are vastly different from those that appeal to adults or seniors.

We added a page for upcoming events on main events page. This is important because we want to display the events that they can attend in the near future. It wouldn't make sense to display past events because that they couldn't attend those events anymore, and it doesn't make sense to display events far off into the future because the user might forget about them. So, to display the earliest events is ideal for an optimal user experience with the kiosk. Our print feature allows the user to get a copy of the event they have just seen on the kiosk, keeping them informed about it even after they have walked away. The printout will include the title, date, website, and details of the event the user has chosen. This means that the user

can look up further information online about the event through the website given, and they can keep the printout as a reminder on when/where the event will be.

In addition, we added a 'Clear All Filters' function, which clears out any existing filter that is currently active. For example, let's say you choose to filter by area and have it set to Northeast, and move throughout the application. If you wish to see the listing of Upcoming Events again, selecting 'Clear All Filters' will automatically jump back to that list. It's convenient due to the fact that you don't have to navigate back through the Area Filter and unselect Northeast to see all the events again.

Finally, our vertical prototype integrates more color and animation. This rationale comes from the user evaluations that we had, where they mentioned all our pages needed more colors, and commented that the visual interface can be more appealing. As a result, building off from the main menu of the horizontal prototype, we included a gradient background between blue and red. We chose blue because it complements the color of the landscape of Calgary as shown in the homepage banner, and a shade of red to complement the blue that we chose. It also represents the Calgary Flames, a professional ice hockey team based in Calgary. Additionally, a set of pictures stacked alongside each other will gradually move from the righthand side of the screen to the left with a looping animation, adding aesthetic value to the application. When you navigate to the Socializing and Events page, headers will have a white font color laid out against a gradient between red and black. Integration of popups in terms of filters, instead of going to a different page for filters like what was previously done in the horizontal prototype, was also done in this vertical prototype through animation of Blend.

SELF-ASSESSMENT OF QUALITY

The current prototype is a decent enough interface for an event selection program, which is one of the pillars of our design. The most useful aspect of it is its modularity; given the similarity of use between our Events page and our Locations page, it would be fairly simple to re-tool the interface so that we could use it to list locations instead of events. Additionally,

the fairly simple palette of red, blue, black, and grey colour gradients matches well with the city's colours and logo.

We also took great pains to retain the simplicity of design that was present in our previous prototypes. It became more difficult as we needed to introduce information, but we managed by keeping all relevant information for an event compartmentalized together. The visual segregation of displays helps to draw the eye, and we followed techniques to match reading behaviour so that users will naturally see the filtering panels first.

Of course, there are drawbacks to our design; data storage and access is difficult to set up on a schedule, so all data access has been simulated as it would appear in a final version. This means that we are drastically underestimating the load times between transactions, which we have not expressly represented in our model. Our current belief is that this load would take place entirely during the fade-out of one list of events, and the fade-in of the new listing would not be delayed.

A second issue with the state of our prototype is the nature of our filters. Currently, we operate on a one-filter model, with a new filter selection replacing the previous one. We have not implemented a multiple-filter model yet, as the lack of a working back-end means manual creation of each potential filter combination. A related issue we are considering for the next version of the software is filter re-labelling, for better recognition of what is currently being displayed (most useful for when multiple filters are applied simultaneously).

Perhaps the most major issue for our prototype is that of feature blocking; currently, we are only allowing entrance into the single section of Socializing and Events. This allowed us to focus our attention on that single section, but also means we have great stretches of content using outmoded graphics.

HEURISTIC EVALUATION PROTOCOL

We could not find recruits, so we chose ourselves as the experts of usability heuristics. As we could not find a specific list of heuristics intended for kiosk analysis, and our application is intended for general use, our team used Nielsen and Norman's 10 Usability Heuristics. From there, we did the evaluations individually, finding issues with the program and writing entries into a table listing them. Each entry states the heuristic violated, the description of the violation, and what needs to be done to solve this issue. We were given a day to complete

this task. Afterwards, we came together and compared the list of violated heuristics we each compiled, then removed any duplicate heuristics and aggregated the solutions found for similar problems. From there, we combined everything into one big table, adding an extra column for severity ratings. We made copies of this table, distributed one to each member, and gave ourselves 10 minutes to individually rate the severity of each entry of the table. Afterwards, we compared the severity ratings of each entry. If everyone happens to have the same severity rating for an entry, we finalized the rating for that particular entry. If at least one member has a different rating compare to the rest of the team, that member has to give a rationale of why they chose the rating for that particular problem area, and one of the other members of the team give their rationale of the rating. After some discussion, we come to a consensus of what the final rating is. We repeated this step for each entry of the table.

RESULTS OF HEURISTIC EVALUATION

Heuristic Violated	Description	Proposed Solution	Severity Rating
Visibility of system status	No load indications between filter selection and event load; this will cause issues for larger selection of events	Add a load indication for the purpose of informing users the request is being processed	2
User control and freedom	When in the Events page, user cannot go back to where they started or previous state	Implement a back button that allows the user to go back to the previous state of the application	2
Consistency and standards	Clear all does not uncheck free events	Ensures that the clear all button resets all of the filters including free events	2
Consistency and standards	'Show Only Free Events' filter is a checkbox, and is not a button like the other filter selections. Causes confusion with the presence of a white square box	Use a button instead of a check box	1

Consistency and standards	Area Selection prompt selection: selecting an area creates a white selection indicator. This is not used anywhere else	Use the same blue shading and border as the other filters. Keep the colour scheme used consistent	1
Error prevention	Print button automatically prints even on accidental selection	Allow a confirmation window to appear before printing the event	3
Error prevention	While a filter selection screen is displayed, users can select a button from the main header which will switch the windows display when the user unintentionally presses the button	Disable the button selection for the main header when a filter is being displayed.	3
Error prevention	While a filter selection screen is displayed, users can select the Print this Event button in the panel behind it	Disable button selection for the events panel while a filter is being displayed	2
Flexibility and efficiency of use	Users- both novices and experts, go through the same interactions. There are no shortcuts available for experts especially for frequent actions.	Have gestures that will correspond to a particular action	0
Aesthetic and minimalist design	Events in listing are not a consistent size and are hidden within the scroll bar	<p>Show all event titles on a single screen; let the rest of info be shown when the event is selected</p> <p>Adjust the sizing of the event so that they only take up minimal amount of space as needed</p> <p>Have a “see more” button, allowing user to see more information (of the description) if they choose to.</p>	3

Help users recognize, diagnose, and recover from error	The print popup is cut off depending on the position of the scroll viewer	Let the print popup to appear outside the scroll viewer.	3
Help and documentation	No help and documentation is found in this application	Have a help page within the application that tells the user what the program is about and how to use it	2
Consistency and standards & Error prevention	Selecting 'Events for Teens' blanks out the events display panel	Check the information allocated to Events for Teens and verify it is correctly configured before published. Grey out the selection if there aren't any available events to prevent selection. Display a message to notify the user there aren't any events for teens	2

From our heuristics evaluation, we concluded that we should need to be more consistent in our prototype and should be more aware of error prevention. During our evaluation, these heuristics were most frequently violated followed by the aesthetic and minimalist design heuristic. Our most severe ratings were problems that are categorized as an error prevention violation and aesthetic and minimalist design violation. The main problems that we encountered would greatly improve the usability of our system by restricting the user on certain pages such that it would prevent unintentional clicking and it would also improve the system by clearly representing the state in a minimalist manner. This would allow the user to understand the system without feeling overwhelmed or confused. The majority of the issues raised were considered as minor problems which would cause disturbances to the usability of the system, but not interruptive ones. In summary, we need to find better ways of showing the user information in simple ways, especially when not doing so has potential to cause errors.

LOOKING FORWARD

Our first implementation necessity, based on what we have done so far, is to fix the major issues found during our heuristic evaluations (the addition of a print confirmation screen, the disabling of inputs when filter selection popups are displayed) and a couple of the minor issues raised from the inspection. Then, we have two directions in which to take this project:

- Deepen the scope of the Events page by implementing a database back end, allowing us to add even more features to this specific element of our overall kiosk design
- Widen the scope of our current interface, and bring the aesthetic and working design to the rest of the areas from our horizontal prototype

At the present time, there appear to be benefits to proceeding in either direction, but by considering the path we've already taken, it would seem that the second option would be more beneficial for our project. Take our current design aesthetic, apply it to the remainder of the pages from our prototype, and then present it to another set of representative users to determine how best to proceed. Through this feedback, we could determine what information we should display on the various pages, which would in turn influence the design of our data storage systems.

As far as further development beyond the next stage goes, we foresee a working printer being a necessary step; this would support our ability to provide users with information even after they leave the kiosk. Another extremely important feature for the domain of our problem would be a language selection prompt; this would best be accomplished once we've reached feature completion, and determined what would be present in the publicly-released version. Given that one of our categories for locations (Trending) and our events page need to change over time, we would also need to develop some type of automated or user-provided system for collecting these listings. Manual curation would be best for a short-term project, but this is one which is intended to stay in one spot for long periods at a time. Finally, we would want to implement some form of gesture-based controls; many users attempted to use our pages as if they were being displayed on a tablet, with tap-and-flick to scroll. This would be valuable for people who have used similar interfaces before, but we would likely also include the original scrollbar control as a backup or for users unfamiliar with the technique.

Appendix

PORTFOLIO 1 CONTENTS

WIREFRAME PROTOTYPE SCREENSHOTS

‘HORIZONTAL’ PROTOTYPE SCREENSHOTS

‘VERTICAL’ PROTOTYPE SCREENSHOTS