

Project Part 3– System Prototype

Project Description

Our project hopes to make events on campus more accessible to Georgia Tech students. These events would include free food availability around campus, rush timings for fraternities and sororities, volunteer opportunities, career fairs, information sessions, seminars etc. Our project would involve the extensive use of Google Glass. We came up with three design alternatives, each unique in its own way to accomplish our goal. We also presented them in class and received a lot of great feedback in which we have incorporated into our conclusion section.

Requirements

We will use Fluid UI for prototyping and Balsamiq for the mockups for the visualization of information.

Design Summary and Usability Specifications

We believe that this project would be well received on campus especially because students remain unaware of a large portion of events on campus. Students may also decide not to go to an event simply because it would take too much of their time to look up the timings and whereabouts of the event. With our project we endeavor to compile all event location and timings into one database. Our application would also provide real time notifications about events as students walk by event locations.

1. Easy to use: Our pop-up and more information views are very easy for novice and experienced users to use.
2. More information: Users have the opportunity to read more about events that they are interested in. They can learn more about event descriptions, food availability and timings.

Initial usability specifications of the application and benchmark tasks it should achieve would include:

- 1) Pop up notification should appear within X yards(according to user preferences) of an event
- 2) User should be able to select and scroll through events in less than 10 seconds
- 3) Events for each day will be refreshed daily and pulled from a database
- 4) Benchmark :
 - a) App to ping when close to a building
 - b) Develop the app to the extent that user might be able to select which events they would like to attend, for example if they would not want to attend any event without free food and the like.

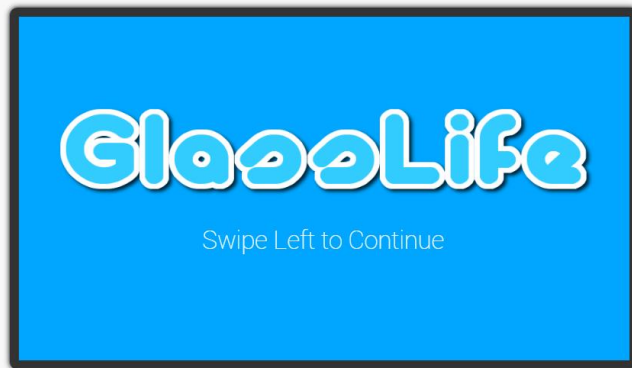
- c) Clicking on the 'more info' tab when an event pops up on the glass so that the user can get more detailed information about an event

Detailed Prototype Demo and Description

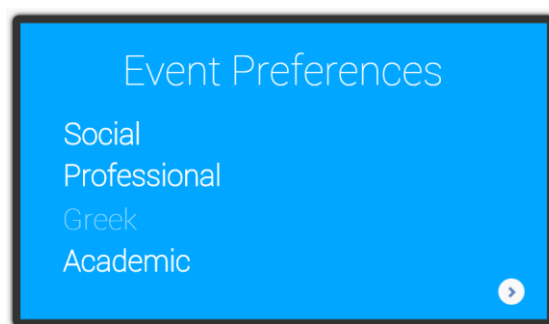
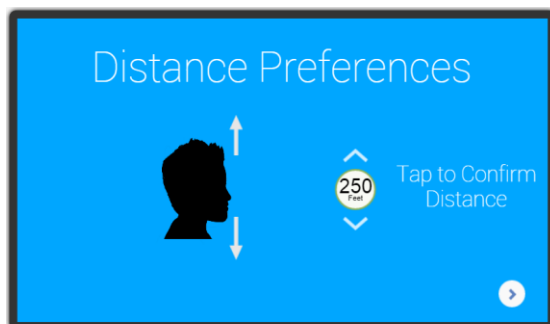
We implemented our prototype using an online prototype, Fluidui.com. By designing first with this platform we were able to visualize each process of the user flow. Initial implementation challenges we faced included finding a proper prototype platform with as well as an unresponsive Glass software. To find the most appropriate tool, we filtered our search results on which tools provided Google Glass interface and commands. However, even after settling for the fluidui prototype platform, we have discovered that the navigation interface does not work well with actual Google Glass testing. For example, to navigate to another slide, Glass relies on a slide action whereas fluidui was designed to have buttons pressed. Developing on Glass itself. initially posed a problem as our computers would not recognize Glass, an issue currently being resolved. .

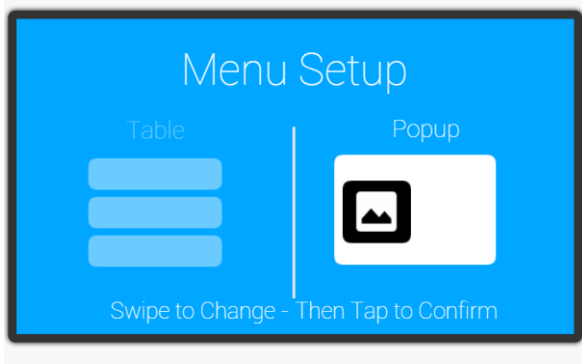
Ideally, our fully functioning version would pull real-time information from the user's location as well as information provided by Glass's sensors. Instead of this real-time data, we created mock situations where a user could hypothetically select their distance and menu preferences.

1) Introduction



2) Settings

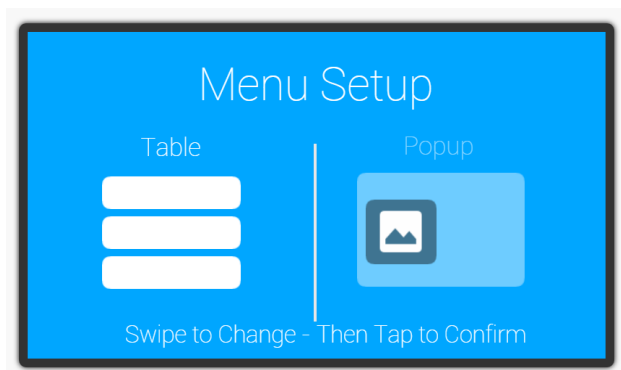




3) Pop-Up view



4) Menu View



Evaluation Plan

Our Evaluation Criteria would include the assessment of the following:

1. Visibility of tasks - Can the user choose which tasks he/she wants to do? Are all possible choices clearly displayed? With our application, the user gets to select his/her preferences as soon as the application is set up. The user would have the choice between a pop up display, where the geographical location of the user would ping and display the event occurring at that location in the closest time frame, and a tabular view where the user could further select which events he would like to go to from all the events listed in the application.

2. Transparency - The system should always keep users informed about what is going on, through appropriate feedback within reasonable time. In our pop up version the user would receive live feedback based on live geographical location feedback. In our tabular view, the events would be listed out based on user preferences and on the selection of each preference the user would receive the appropriate feedback. For example, if a user implies that they would like to only attend free food events then the application would ask if they were sure about their decision and if they would like to ignore certain pertinent events like information sessions.
3. Feedback Mechanism - The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. The system should incorporate natural mapping where possible. Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution so that even novice users can be able to use the application without much difficulty.
4. Ease of Remembrance and Recognizability of Instructions- Instructions for use of the system should be visible or easily retrievable whenever appropriate. The user should not have to memorize a certain set of instructions. The user should be able to view all appropriate instructions again or in our case revisit their settings and refresh them.
5. Relatability of the Application - The system should understand and speak the users' language i.e. with words, phrases and concepts familiar to the user, as opposed to system-oriented terms. The application should also follow real world conventions like making backgrounds have a high transparency or making information appear in a logical manner.
6. Controllability. Navigation and Ease of Use - Users often choose certain functionality by mistake, thus our application should have a clearly marked 'Back; or 'Undo' functionality. In our case, we will have a set of preferences that the user can edit at any given time.
7. Consistency of Information - Users should not have to wonder whether different words, situations, or actions mean the same thing. Our application aims to follow platform conventions to maintain uniformity. All background and text colors should be relatively consistent throughout the application. Most pictures would remain uniform in size and the passive or active voice, whichever we choose to implement, will be consistent throughout the application.
8. Error prevention - Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action. In the event that errors do occur, error messages should be

expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

9. Aesthetic and minimalist design - Our application design will be streamlined with basic colors and serif and sans serif font. Our application aims to be as concise as possible.

Further synthesizing the above criteria, we developed questions that we hope any user of our application would be able to answer. These questions serve as a critique of sorts so as to provide us with insight into what we need to improve upon. These questions would be along the vein of:

1. Are the instructions at the beginning of the application understandable? Is the option between whether the user selects a pop up version or an application version intuitive i.e. can the user understand the choice well enough to make an informed decision?
2. Is the 'More Info' tab visible? If not, would a 'More Info' tab at the top of the page grant the user better access to more information as opposed to the current format with buttons for each event?
3. Was our application easily navigable? Was more information required about which action corresponded to which aspect of navigation i.e. when to slide and when to tap?
4. Our feedback mechanism for the categories of events the user would like to view was to select or highlight required fields and fade out fields that were not selected by the user? Would you prefer say a set of check off boxes instead?
5. How transparent was our application? Could you tell how certain actions affected the final application? For example, setting the distance of the application text on the glass.
6. Was the reversibility of actions clearly displayed? Did we have an undo button?

Summary and Reflection

The main aim of our design is to reach out to the whole student body at Georgia Tech and help them get involved in Campus Life, thereby building a strong community on campus. We aim to notify students about the various professional, academic and social events on campus so that they are aware of these events and can attend them based on availability.

We choose a mix of this pop up and tabular view for our application as it is one that can be easily viewed and implemented on Google Glass. As the screen space is limited on Glass, we wanted our application to be concise and with adequate functionality. We could have chosen to include, say a calendar in our application so that the user could mark out when they had the time and inclination to attend events but we realised that the tap and slide functionality of the glass was not conducive to this feature. Thus we removed it. Our application was designed after much deliberation on which design would provide our users with optimal performance while at the same time not compromising functionality. Our application will prove extremely useful to students and even faculty at Tech, in that it could become a common area in which all event notification could be posted. Faculty and student organizations could send in their events and this

application could serve as a means of advertisement to familiarize students with newer organizations and their events. Free food events are mostly spontaneous i.e. not enough awareness of the event can lead to its failure. Our application will also allow students to catalog all the events they want to go to according to their preferences, for example students can choose to go to only free events or events after 6 pm. This is a great way to get involved with the university community in an easy and user friendly manner.

In addition our design intends to follow the following parameters for success:

- 1) Recognition: When user has the app open, google glass should be able to recognize nearby building within 500 feet as users walk around campus
- 2) Search Time: Once the building is recognized (via image or wifi signal), a query database search for events happening now and nearby future of that building should take no more than 5 seconds.
- 3) Food Availability: App should have a sign to tell the user if a particular event has free food or not.