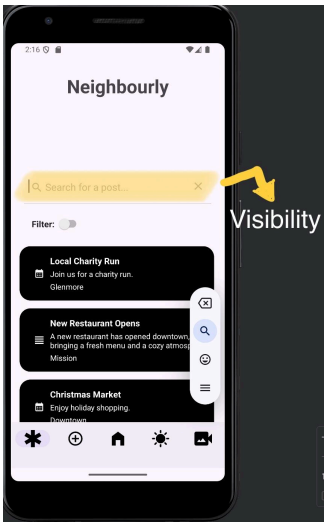
COSC 341 Project Step 4

Group Number: 24
Team Members: Juan Ablan (23973282), Madelyn DeGruchy (85064962), Elana Wood (79393492), Tatum Grundy (65316812)

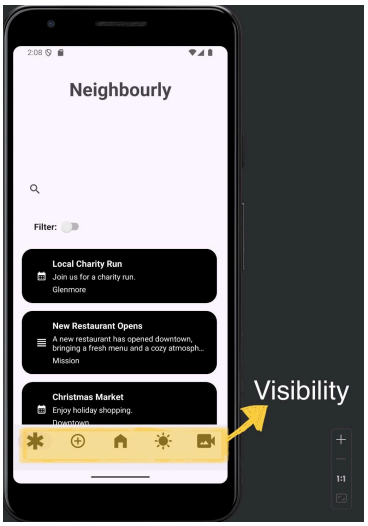
System Overview

The developed system is a Local Community Communication Application called Neighbourly. Our community platform allows residents of a particular city to post about community events (in their neighbourhood or city-wide), connect with people in their area, and get information about key alerts (safety concerns, weather events, power outages, etc.) all in one place. The four major tasks of the system are; 1. Creating, deleting, and editing posts and events, 2. A feed displaying posts with filters, search view, and scroll view, 3. An emergency alerts dashboard, and 4. Weather and webcam pages which display the current weather and forecast, as well as local webcams.

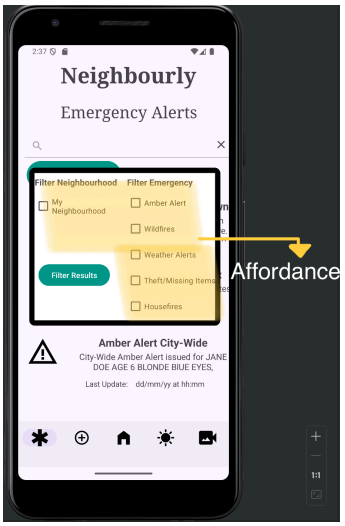
Design Principles



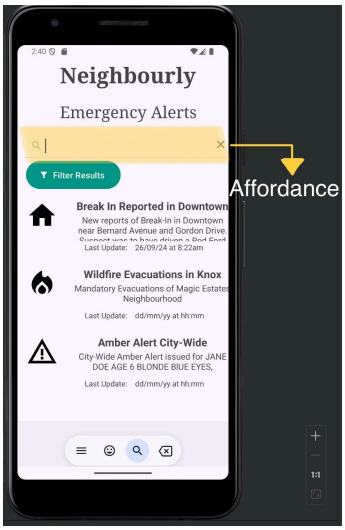
Search functionality is always active on the main feed page ensuring that users can loop up content without friction.



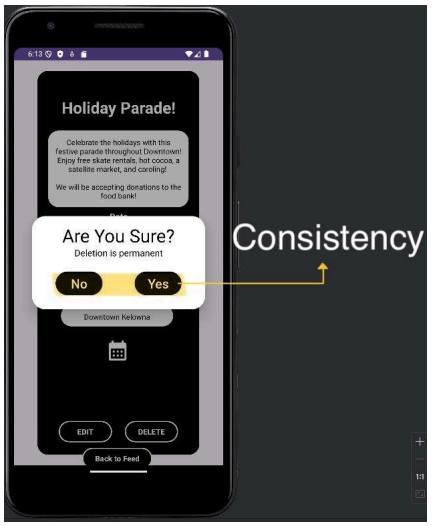
Tab Bar Menu integration improves the visibility of users' key experiences.



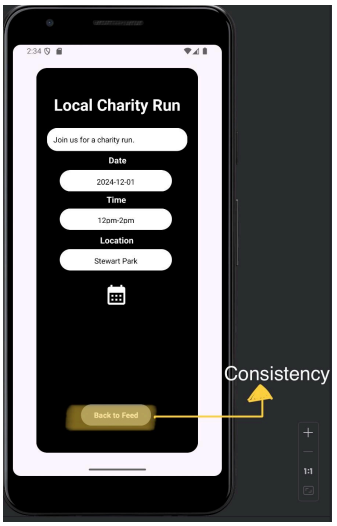
Filter checkboxes being square multi-select options indicate to the user that they can make multiple selections from either column (Neighbourhood and Type)



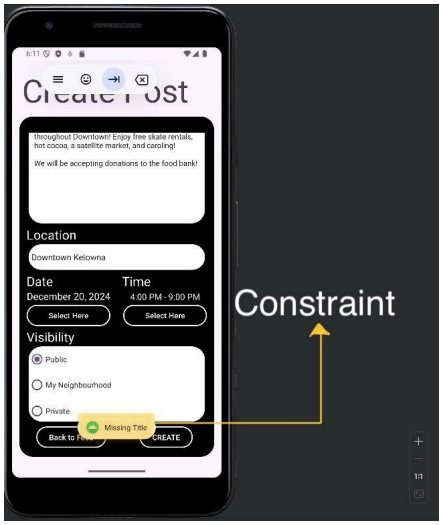
Search Bars include a magnifying glass on the right, an x to exit the search function, and at launch, the search box has a blinking cursor to indicate a text field to indicate to the user that this is a text field



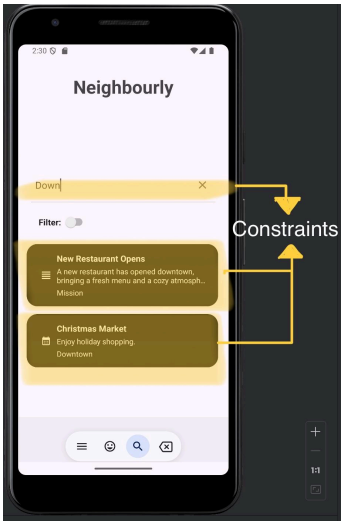
Any confirmation dialogue always uses the words “yes” or “no” to gain the users’ consent before committing the action. They always appear on the same sides, with “No” on the left and “Yes” on the right.



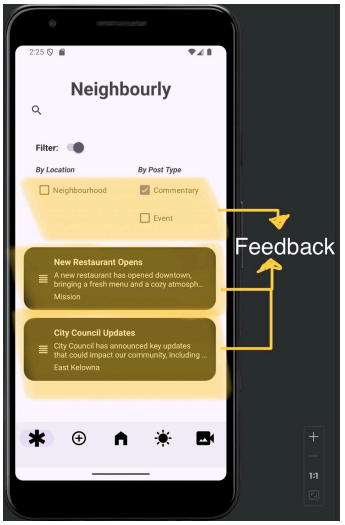
Back to Feed or return buttons are always in the centre at the bottom of the screen when you enter a secondary page.



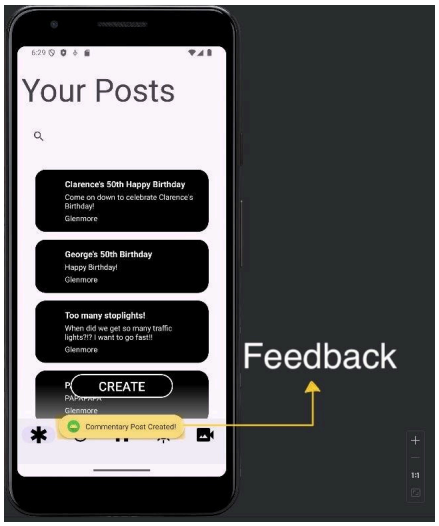
The app forces the user to enter a complete input and will provide a toast message indicating what was missed.



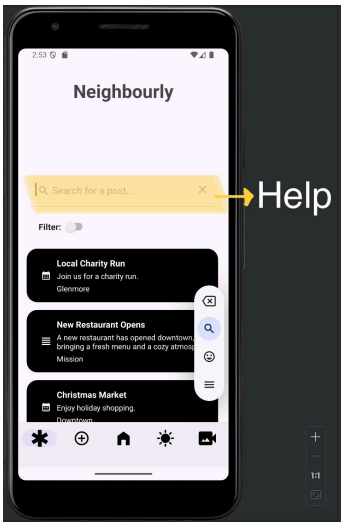
When searching for a post throughout the app, user input receives instant feedback without needing to submit their search.



When The user toggles a filter option, it is filled to indicate it has been selected and the filtered results are shown immediately.

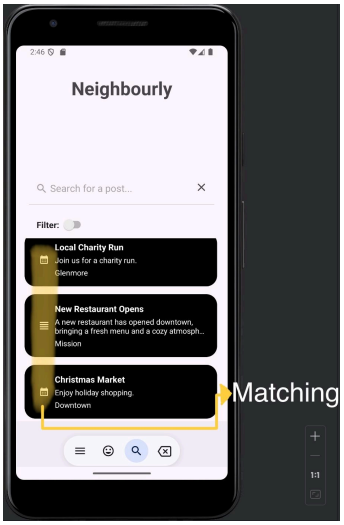


After creating a new post, a toast appears to let the user know there post was created successfully. It also indicates the type of post again to the user.

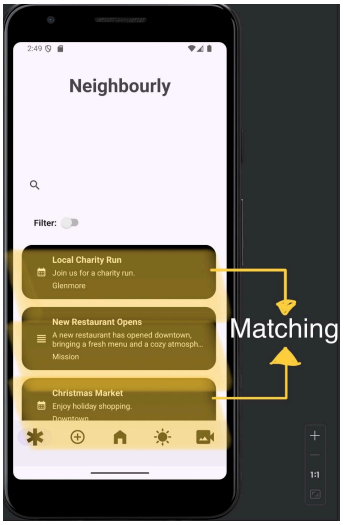


Hint text in the search bar helps users understand where and what to type to properly search for the content they are looking for.

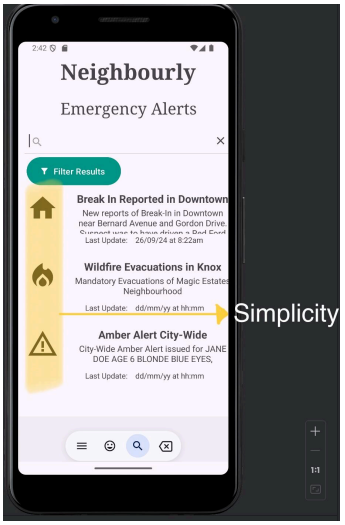
Placeholder text shows users what kind of information needs to be entered to succesfully create a post. This text also shows the user where they need to input their details.



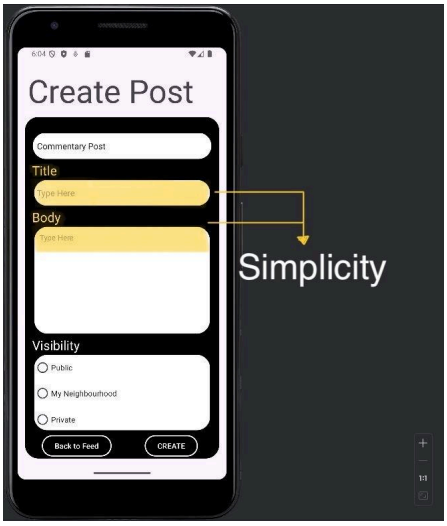
Icons are logical matches, the calendar indicates an event that occurs at a date and time that matches what people expect a calendar to be.



The feed layout matches common social media platforms, with the search bar and filter options near the top and a vertical scroll through posts.



Icons are basic and filled in black or white to easily identify post and emergency types. They are set aside in an uncluttered area for users to discern the post type quickly.



When creating a new post the user is only shown the inputs for the relevant information for the post type. Irrelevant details are hidden; the date and time needed for event posts are hidden from the commentary view.

Heuristic Evaluation

A heuristic evaluation was conducted for the system in which three members from other project groups used Nielsen's heuristics to identify aspects of the interface that violated aspects of the heuristics. The prototype was updated based on feedback from heuristic violations, these changes are detailed in the Updated Prototype section. The full results from the individual heuristic evaluations have been included in a separate PDF file. A summary of problems identified by each user, along with which heuristics were violated, severity, and user's recommendations is provided in this section sorted by each heuristic violated.

Match Between System and Real World

User 1 identified that the separate webcam activity that opened after an image was clicked did not properly open the webcam, and a confusing network error appeared in the embedded webcam Webview in the OpenWebcam activity. The severity of this violation was 3, indicating a major usability problem. User 1 recommended that the webcam be opened in a web browser rather than embedded into an activity so that the error messages received would be less confusing and technical sounding.

Aesthetic and Minimalist Design

User 1 noted that there was a lack of detail and cosmetic issues in the weather forecast, and recommended making the design more cohesive by sticking to a specific colour palette. User 2 identified that the text in the emergency post section was cut off, and recommended that the width constraints for the text views be updated so that there would be enough of a margin between the text and the edge of the screen. User 3 identified that there were inconsistent style choices between some functionalities, and they recommended that one font and colour for design elements be used for the application. The severity of these heuristic violations were all scored at 1 (Cosmetic).

User Control and Freedom

User 1 noted that there was no way to go back to the application after opening a webcam URL, they recommended providing a clear way for the user to go back to the application via Toast message with brief instructions or a back button. User 1 also noted that there were no back buttons for the post create and editing pages, they recommended that back buttons be incorporated into the post create and editing pages to create an easy-out for the user. The issues that User 1 found were all scored at a 2 (Minor).

Visibility of System Status

User 2 identified that after selecting any of the pages from the bottom navigation bar, the highlighted selection always returns to the emergency alerts tab once the page has opened. They recommended updating the selection to reflect the current menu item being viewed. This issue was scored at a 2 (Minor).

Consistency and Standards

User 2 noted that the different filter options work differently within the app, one opens a pop-up and the other opens within the page. They recommended that one of the layouts be modified so that both filter views are the same in the main feed and the emergency dashboard. This issue was scored at a 2 (Minor).

Help and Documentation

User 3 identified that minimal help and documentation are provided in the create post fields, they recommended that tooltips or toasts be added near the fields to explain what each field is/does. This issue was scored at a 2 (Minor).

Recognition Rather than Recall

User 3 noted that there are unclear menu items as they do not provide text labels for what they are/do. They recommended that titles below each icon be added so that it is clear what icon takes you to what menu item. This issue was scored at a 2 (Minor).

Vertical Prototype

A fully functional vertical prototype of Neighbourly which addresses the four main tasks was developed in Android Studio. The source code for the application has been submitted along with a video sharing how the system is expected to be used for the tasks. To access the video posted on YouTube, please click this link or the one provided as a comment on our submission.

Updated Prototype

The prototype was updated based on feedback collected from the heuristic evaluation. The individual heuristic evaluation that scored a severity of 3 indicated that after a user selected an image in the Webcam Activity, the code indicated that a separate activity called ‘OpenWebcam’ was meant to open with an embedded WebView containing the URL of the selected webcam. When a user selected an image, the OpenWebcam activity was launched, but the embedded WebView displaying the user-selected webcam displayed a network error and not the webcam. The error message could not be easily understood by the user when viewed in the embedded WebView. The changes that were made to address this feedback included changing the Webcam Activity’s implementation of the WebView so that it was launched in a browser instead of embedded into the application itself. The openWebcam activity was deleted and the handling of the image button presses in the Webcam activity was modified to launch the webcam URLs directly from the user clicking an image. In addition to this, heuristic feedback indicated when a user was creating or editing a post, there was no way for them to navigate back to the previous page if they did not mean to create or edit posts. To implement this feedback into the prototype, the edit and create post java activities were modified to add buttons that navigate back to the previous page that the user was on (the my post view), this allows users to have more control over the application.