SCRIPT

Hello everyone, we are team StrayFinder from DSS3. Today, we’ll be demonstrating how our app works, but before that we’ll be briefly describing what this app is about. So the app we made is basically a one stop platform to facilitate the ease of reporting and tracking the whereabouts of stray cats for both passersby and volunteers. The passersby can easily report the cat they found through the app, including its injuries, and allow volunteers who are using the app to get updates of where the cat is last seen so that they will be able to locate, and rescue said cat. With that being said, our app has 2 main users: passersby and volunteers.

Now, I will demonstrate how the app works in the perspective of a passerby who just happen to find a stray cat on the streets. Let’s call the passerby Kevin. Because Kevin is a passerby, he clicks on the first button for passersby. Then, Kevin will be prompted to upload a picture of the cat, either through gallery or through the camera. For the purpose of this demo, Kevin will be uploading the picture of the cat through his camera gallery. However, because Kevin wasn’t looking carefully, he accidentally uploaded a picture of a dress instead of a cat and pressed the green button to proceed to the next page. However, because our app is able to detect if the picture uploaded is a cat or not, there will be a pop-up error message that tells Kevin only cat pictures are allowed. So now, he reuploads a picture of the cat he found on the streets. While we wait for the screen to load, I will describe what the app is doing right now. What it’s currently doing is to predict the breed of the picture of the cat uploaded and once the breed is determined, the app will fetch pictures of cats with the same breed in our database. Then, it will prompt Kevin to try and match if the cat is an existing cat in our database or is it a new cat. That is why all of the pictures here are of the same breed as the picture of the cat uploaded by Kevin just now.

For the purpose of this demonstration, let’s say Kevin thinks the cat he found is a new cat. Then, he will have to name the cat he found, and, in this case, he decides to name the cat Andy. After he confirms the name, he is asked by the app to allow the app to gain access to his location. This is done so that the app can update the location of Andy as the current location of Kevin. After allowing access, Kevin is asked by the app whether Andy is injured. Let’s say Andy is injured, so after Kevin confirms the injury, he is prompted to upload pictures of the injuries to the app and write a brief description about it. Afterwards, he is also asked if he is going to bring Andy to the vet himself or not and if he indicates yes, he will have to leave his personal contact details so that a volunteer will be able to take over the case. After filling up and confirming his contact details, there will be a thank you page that indicates the end of the reporting process and if he presses the ‘report more cats’ button, he will be redirected to the first page again.

Let’s say, after reporting Andy, Kevin wants to adopt a cat, but he doesn’t know where to go or who to contact. Our app also has an adopt tab that allows Kevin to browse through a list of registered adoption agencies in Singapore, so he’ll be able to contact the organizations directly. There are also hyperlinks to the adoption agencies’ websites so if he clicks on it, he will be redirected to the website.

Let’s say, after agreeing to bring Andy to the vet himself, Kevin wants to know where the nearest vet to his current location is. He can find that out by accessing the vets tab, which will show all registered vets in Singapore on a map. The data about the vets is taken from govdata. For example, if he wants to find out details about the vets, he can easily just press on the marker on the map and an info window will show the full details of the vet. He will also be able to click on the ‘navigate using Google Maps’ button after pressing on any of the markers so that he’ll be redirected to Google Maps to get directions on how to go to the location of that particular vet. To filter to the 5 nearest vets to his location, he can easily press the floating button on the bottom right and press the ’filter 5 nearest vets’ button and only 5 markers will show up on the screen. To go back to the original map with all the vets, he can just press the floating button again and press the ‘show all vets’ button.

Now, I will demonstrate how the app works in the perspective of a volunteer with the name of Sally, who works under SPCA. Before logging in, Sally needs to register a joint SPCA account in our app so that all volunteers in SPCA will be able to use our app without making multiple accounts. First, she filled in all the details needed to register an account such as the organization name, the email address, the password, the contact number, and the physical address of the organization. After registering, she will be able to log in to the account she just created. However, somehow the password she just created slipped her mind and she could not remember it. But no worries, because our app has a feature to allow her to reset the password through email. So, she just needs to click on this button, which will direct her to a page where she’ll be able to input the email address she previously registered under. A procedure on how to reset the password will then be sent to the email and she will be able to access an external link to reset it. After resetting, now she is ready to login to the SPCA account by entering the email address and the new password.

Once she is logged in, she will be able to see a list of all injured cats in our database, along with pictures of the injuries and other details of the cat. Here, she will be able to see Andy, the cat Kevin had previously reported. She will then be able to the see the location of Andy by clicking on the location tracker, which brings her to a map where she will be able to get directions via Google Maps to the location of Andy. She will also be able to contact Kevin since Kevin decides to bring Andy to the vet himself and leaves his contact details. Let’s say Sally has handled the case and has successfully taken Andy from the vet. Because Andy is no longer injured, Sally can just press on the handled button to ensure that Andy is no longer registered as an injured cat in our database.

Let’s say Sally wants to know the location of all the cats in the database. She can easily do so by accessing the cat map tab, which shows a map with markers indicating the last seen locations of all the cats in our database. The markers for injured and non-injured cats are also different: white cat marker is for non-injured cats whereas purple cat marker is for injured cats. She can also easily see the details of the cat in an info window by pressing on the marker and she can also get directions via Google Maps to navigate to the location of the cat.

Similar to Kevin, Sally also has access to the vets tab, where she’ll be able to know the exact location of all registered vets in Singapore. She’ll also be able to find out the nearest 5 vets to her location by clicking on the floating action button and the filter button consecutively. To return to the map with all registered vets, she can just click on the floating action button again and click on the ‘show all vets’ button.

Let’s say Sally wants to the edit the SPCA’s account’s registered physical address. She’ll be able to do so by clicking on the user icon on the top right-hand corner and change the organization name, the contact number, or the address of the account. To edit, she just needs to click on the edit icon at the top right-hand corner and once she is done editing, she can just press confirm and the changes will be automatically reflected afterwards. To log out of the account, she just needs to press on the icon button next to the user icon and she will be directed back to the login page.

That is all we have for our app demonstration. Thank you!