Snap + Seek – Mobile Interaction Report

Background

Snap + Seek is a mobile application that uses the device's camera to play a game. The application provides the user with an item to find and to take a picture of within a certain time frame. If the picture matches the item, then a point is scored and time is added to the timer. The goal of the game is to find as many items as possible before the timer ends.

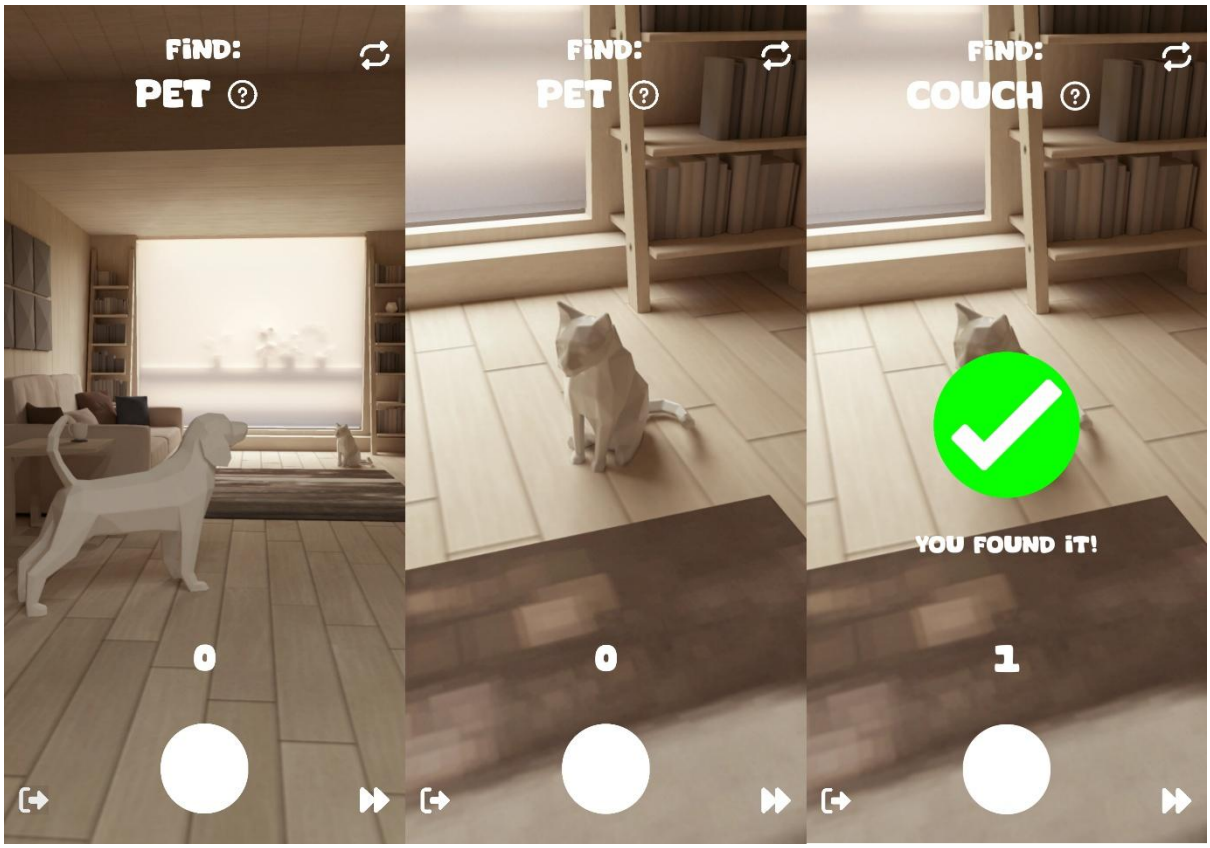

Figure 1. View of the main user interface

The game requires the player to physically move around their homes and to take pictures of certain items around their house. The application was made for a wide audience and is intended for use inside the user's home. There were many design and technical opportunities presented by the application. In relation to design, the goal was to deliver an experience which facilitated a wide userbase, ensuring that the game is easy to learn and incorporate a minimalist design to facilitate the device's camera without obstructing the view. The application had to be easy to navigate and present information to the user in a simple but understandable way. Technically, the idea presented an opportunity to build on existing machine learning capabilities to deliver a functioning mobile game using the device's camera. Other features could be incorporated, such as use with additional APIs to provide the user with more information and use the device's storage to provide a personalised experience. Existing applications, such as Camera Hunt (Elias, 2018), provides a fairly rudimentary experience which uses similar machine learning capabilities along with the camera to make a game. Snap + Seek intends to deliver a more comprehensive experience, incorporating more features, such as item

descriptions and difficulty selections, to make the game more engaging and enjoyable for the user. In addition, a more modern and aesthetically pleasing design with improved visual feedback will be delivered to improve the user experience.

Description and Justification

Activities and Design

The application consists of 5 different activities, including Main, Camera, Settings, Tutorial and Game Over. The Main activity allows the user to start the game, go to the settings or view the tutorial. It also displays the player's current high score. The Camera activity is where the game takes place, displaying the device's camera and the game's buttons. The Tutorial activity displays information on how to play the game. The Settings activity allows the player to change the game's difficulty from Free, Easy, Medium and Hard. Free mode allows the player to freely play the game without a time limit, while the other difficulties have varying time limits and adjustments to adding and removing time. The Game Over activity displays the final score of the player and buttons to replay or go back to the Main activity. All activities incorporate conventional UX heuristics to ensure a positive user experience and simple navigation. Buttons on the Camera activity use conventional icons related to their functionality, and loading icons are present to display the system's status. Buttons across all pages are identifiable and theming is consistent throughout.

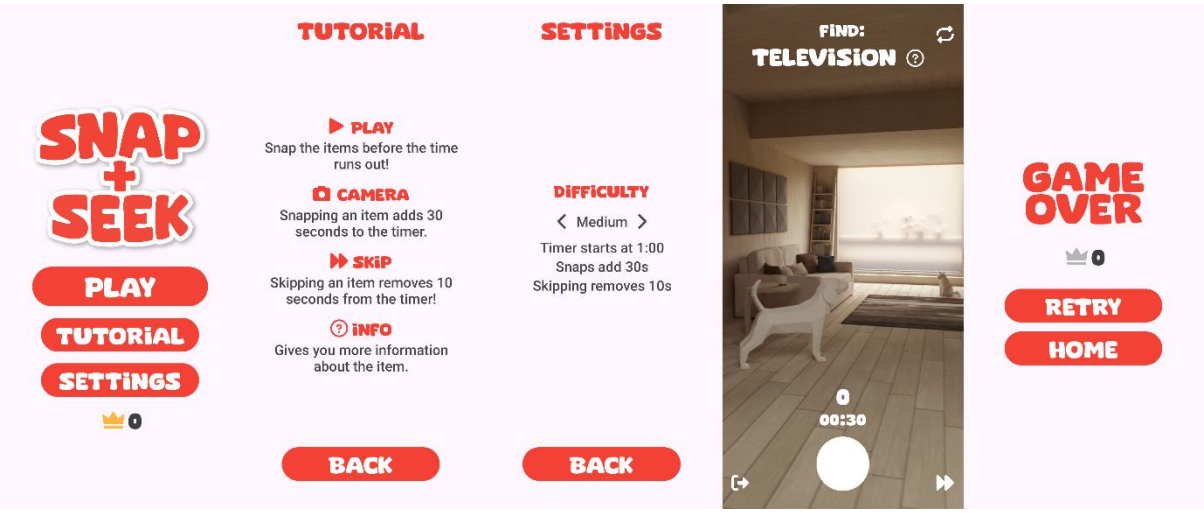

Figure 2. View of all pages

Camera

Before labelling an image, a picture using the device's camera must first be taken. This was implemented with the use of the CameraX library, part of Android Jetpack. For the camera sensor to function, the 'Camera' permission must be requested by the device and accepted by the user. The 'Write External Storage' permission is also used to save images to the device. The layout file 'activity_camera.xml' includes a view finder in the background which matches the screen size of the device. When the camera activity begins, permissions are checked and the camera preview is started. When the user presses the image capture button, a time stamped name and 'MediaStore' entry is created, determining the file's name, type and destination. The image is then captured from when the user presses the button and saved to the storage.

Image Labelling

Once an image is saved to storage, 'labelImage' function is called. The function uses the ML Kit Image Labelling API, which recognises more than 400 labels to classify an item that appears in a given image. The function pulls the image from the drive and generates all the labels above 60% confidence that relate to the image taken. Generating all labels regardless of confidence level was found to be unreliable and could produce incorrect matches, whereas a confidence level that was too high often meant no matches were found, so 60% was chosen as a middle ground. A for loop then checks each of the results and checks if there is a match between the current item the user has to find and the generated image labels.

Game Logic

When the Camera activity begins, the countdown timer begins, with the time limit depending on the difficulty the user selected. The item is then generated by choosing a random integer between 0 and 294, the number of labels in 'labels.json'. The number of labels was reduced so that items can realistically be found inside someone's home. This label is pulled from the file and displayed on screen. When the user takes a picture and a match is found, visual confirmation will appear, with a new item being generated. The score will be updated by one, and time will be added to the timer depending on the difficulty. If there is not a match, then visual confirmation will appear encouraging the user to keep looking. If the user cannot find an item, they can press the skip button. When the skip button is pressed, visual confirmation will appear, and a new item will be generated. Depending on the difficulty, time will be removed from the timer. Once the timer is complete, an intent will load the Game Over activity, passing the score.

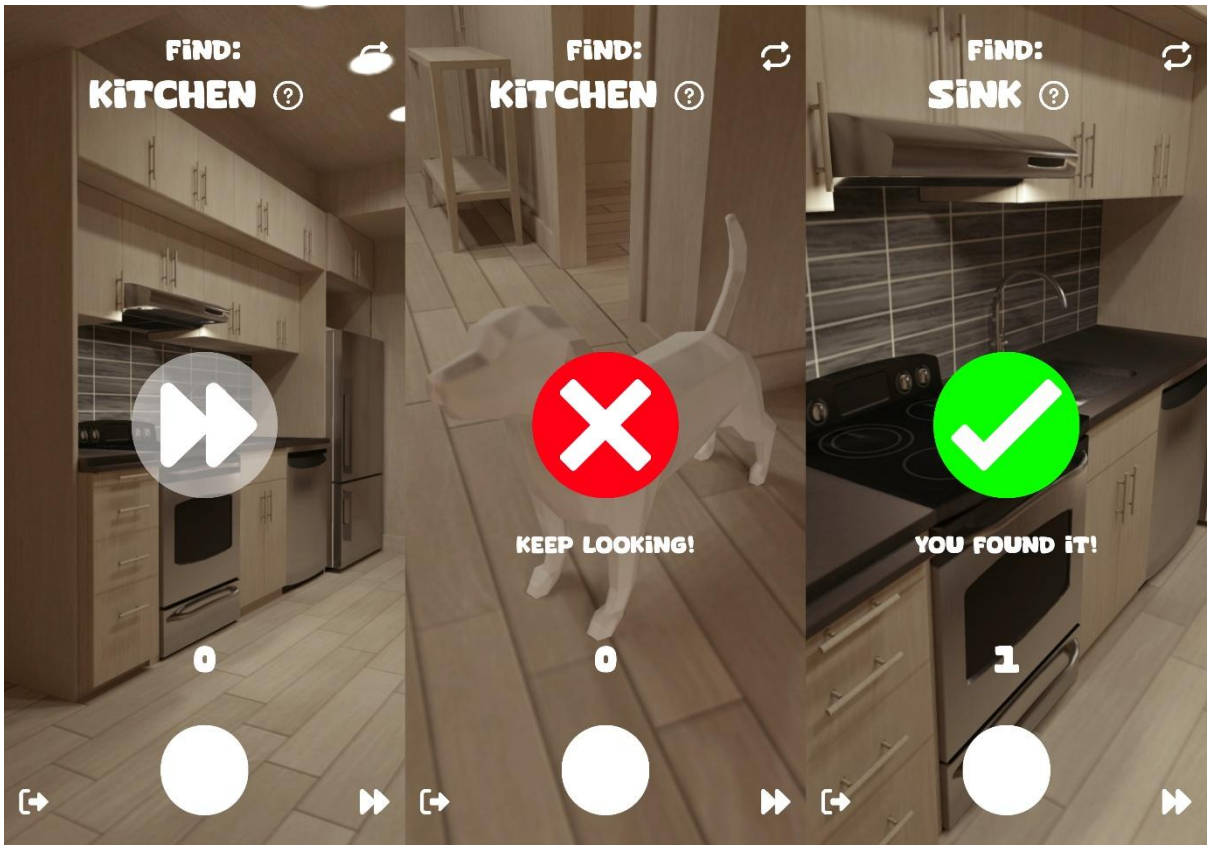

Figure 3. View of the visual confirmations

Item Description

If the user is unsure about a given item or what it looks like, they can press info button next to the item's name. When an item is generated, it's description is also generated using the Wikipedia API. The API is called and the JSON response is converted to a string. If the response contains an image, then the image URL and description is saved. If the response does not contain an image, then the extract is saved instead of the description, as it contains more information, to compensate for the lack of an image. When the user presses the info button, a dialog box is shown, containing the current item as a title, the associated description, an image view if an image is available and an 'OK' button (see Figure 4). If an error occurred loading the item's description, then the dialog will simply show that the item's description is unavailable.

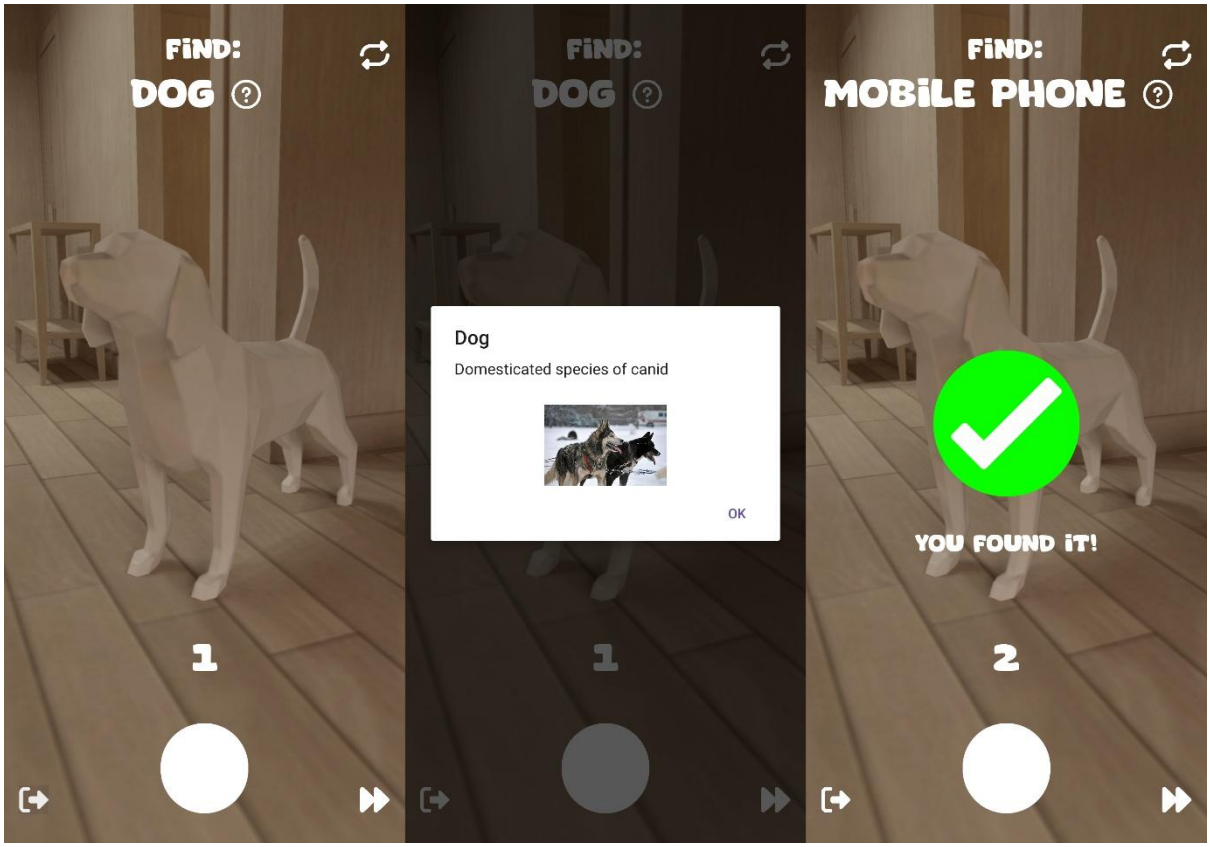

Figure 4. View of the item description dialog

Ethics and Security

The application uses only the required permissions for the app to function. These only include internet access, camera access and the ability to write to external storage. Pictures taken by the camera are solely stored on device and not shared elsewhere. When using the ML Kit Image Labelling API, processing of the image fully happens on-device, and that data and the resultant outputs do not get sent to Google's servers (Google, 2025).

Critical Reflection

The goal of the project was to create a mobile game that uses the device's camera in conjunction with machine learning to deliver a fun and engaging experience. I believe the project largely achieved its aims but could be improved in some ways to further enhance the

application's capabilities. The camera and machine learning capabilities work as intended, delivering a usable and informative user experience while implementing an effective gaming experience. The application is easy to navigate and follows conventional design features with consistent theming throughout. ML Kit's base model for the Image Labelling API does have some limitations regarding the number of categories and the specific labels themselves. By using a custom model, the app could be improved further by providing a greater number of categories that are more specific, such as objects found in a household. The integration of the item description could also be improved. It does function for most items, but some are missing information altogether, usually due to multiple definitions or meanings to certain words. The design of the application could also be improved in order to make it more aesthetically pleasing, such as with background images or more detailed colours. Despite some weaknesses, I believe that the application is functionally complete and provides an enjoyable user experience.

Bibliography

Google. (2025). *Terms and Privacy* [Online]. Google Developers. Last updated 14 May 2025. Available at: https://developers.google.com/ml-kit/terms [Accessed 8 January 2026].

Elias, R. (2018). *Camera Hunt – Scavenger Game*. [App]. Available at: https://play.google.com/store/apps/details?id=com.camerahunt.camerahunt&hl=en_GB [Accessed 8 January 2026].