

KAUNAS UNIVERSITY OF TECHNOLOGY

FACULTY OF INFORMATICS

T120B169 App Development for Smart Mobile Systems

*Fishing Time*

|  |
| --- |
| *IFZm-1, Emilija Kravčenko:* |
| Date: *2024.04.09* |

Kaunas, 2024

Tables of Contents

[Description of Your app 3](#_Toc163597386)

[Functionality of your app 4](#_Toc163597387)

[List of functions 4](#_Toc163597388)

[Solution 5](#_Toc163597389)

[Task #1. Displaying action and navigation bar only on swipe 5](#_Toc163597390)

[Task #2. Navigation between fragments 6](#_Toc163597391)

[Task #3. Reacting to button clicking and image clicking. Remove or change UI component from the activity when a button is clicked. 7](#_Toc163597392)

[Task #4. Moving an image with a finger 8](#_Toc163597393)

[Task #5. Saving game state when flipping screen 10](#_Toc163597394)

[Task #6. Creating a history of each caught fish 12](#_Toc163597395)

[Task #7. Saving all game progress, loading it after closing game 13](#_Toc163597396)

[Task #8. Creating a shop where player can buy new fishing locations 16](#_Toc163597397)

[Task #9. Changing fishing location from a list of owned location 18](#_Toc163597398)

[Task #10. Animating an image 19](#_Toc163597399)

[Reference list 20](#_Toc163597400)

# **Description of Your app**

1. Cozy fishing game
2. Description: Ekrane matoma viena plūdinė meškerė. Kai pamatai kad kimba, iš plūdės elgesio, paspaudi ant meškerės. Paspaudus matai pritrauktą  meškerės vaizdą per vidurį. Gali paspausti mygtuką kirsti ir jei užkimba pritraukinėji laimikį spaudžiant dešinį mygtuką o su kairiu pirštu kontroliuoji meškerę, kaip nori, paslenkant ją ekrane, kuri lenkiasi nuo žuvies. Jei sėkmingai pritraukei žuvį pamatai iššokusiame lange ką pagavai ir gauni pinigų. Už pinigus gali nusipirkti naujas žvejojimo vietas ir keliauti tarp jų. Jei nesėkmingai pritraukei ir žuvis nutrūksta pamatai iššokusiame lange, kad tau nepavyko ir vėl matai vieną meškerę.

# **Functionality of your app**

## **List of functions**

1. Displaying action and navigation bar only on swipe
2. Navigation between fragments
3. Reacting to button clicking and image clicking. Remove or change UI component from the activity when a button is clicked
4. Moving an image with a finger
5. Saving game state when flipping screen
6. Creating a history of each caught fish
7. Saving all game progress, loading it after closing game
8. Creating a shop where player can buy new fishing locations
9. Changing fishing location from a list of owned location
10. Animating an image
11. Adding more shopping items
12. Changing fishing gear
13. Adding music and sound effects
14. Showing pictures of fish from internet
15. Album of all possible fishes

# **Solution**

## **Task #1. Displaying action and navigation bar only on swipe**

After opening the game user does not see an action and navigation bar unless they swipe from the very top down or from the very bottom up in vertical phone view or from the very top down or from the furthest right to left in horizontal phone view. After swiping, the user has a few seconds to use those bars before they disappear.

A fishing float in the water

Description automatically generated

Figure 1. Screenshot #1

A fishing float in the water

Description automatically generated

Figure 2. Screenshot #2

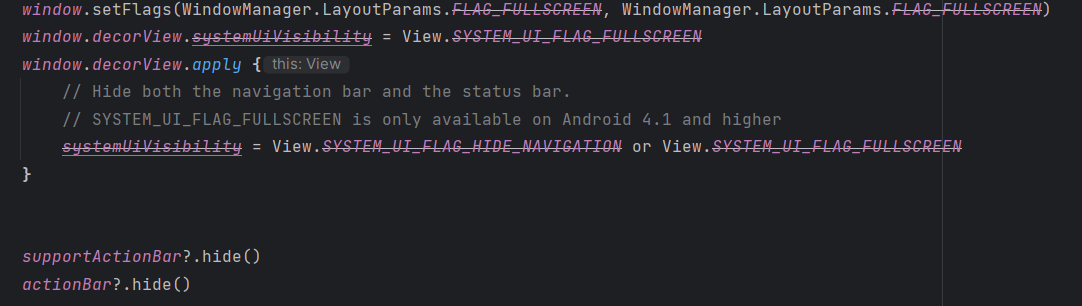


Figure 3. Source code #1

## **Task #2. Navigation between fragments**

Navigation Graph controls movement between fragments. After pressing on a button or an image navigation graph is called to make transition to next fragment. Also automatically works when pressing back button and takes user to previous fragment. The Back button action can be reprogrammed.

A screenshot of a computer

Description automatically generated

Figure 4. Screenshot #3

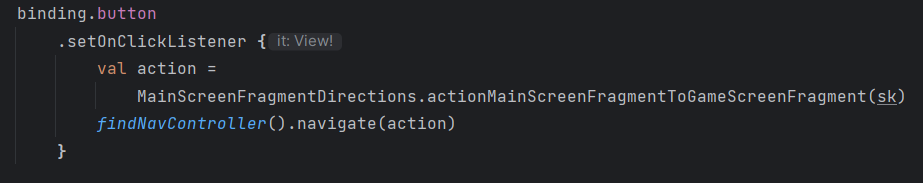


Figure 5. Source code #2

## **Task #3. Reacting to button clicking and image clicking. Remove or change UI component from the activity when a button is clicked.**

After clicking on a button or some images a function is called. For example: Clicking on button “Kirsti” makes it disappear and makes an image appear. Long clicking on reel image calls another function that counts the number of correct reels, rotates image.

A person in a yellow and black pole in the water

Description automatically generated

Figure 6. Screenshot #4

A person in a yellow and black pole in the water

Description automatically generated

Figure 7. Screenshot #5

A computer screen with text

Description automatically generated

Figure 8. Source code #3

A screen shot of a computer

Description automatically generated

Figure 9. Source code #4

## **Task #4. Moving an image with a finger**

After successfully striking the player can move an image of a rod across the screen but can’t move it completely out of sight. Movement area depends on screen size and position. Image position is recalculated after flipping the screen.

A yellow pole in water

Description automatically generated

Figure 10. Screenshot #6

A fishing pole in the water

Description automatically generated

Figure 11. Screenshot #7

A screenshot of a computer program

Description automatically generated

Figure 12. Source code #5

A screen shot of a computer program

Description automatically generated

Figure 13. Source code #6

## **Task #5. Saving game state when flipping screen**

While catching fish, if a screen is flipped, game progress is saved. Visibility, rotation, position of buttons and images, and the number of successful reels are all saved and recreated after flipping the screen. This way, the game remembers if you already correctly moved the rod and lets you reel in flipped view. Code reference was taken from Source#[[1](https://www.youtube.com/watch?v=TcTgbVudLyQ)].

A yellow pole in water

Description automatically generated

Figure 14. Screenshot #8



Figure 15. Screenshot #9

A screen shot of a computer program

Description automatically generated

Figure 16. Source code #7

A screen shot of a computer program

Description automatically generated

Figure 17. Source code #8

## **Task #6. Creating a history of each caught fish**

After pressing the history button in menu screen player can see how much money they have, how many fishes they caught, date and time of each caught fish. Times are displayed in recycle view, using adapter. Player can scroll through times.

A blue rectangular object with numbers

Description automatically generated

Figure 18. Screenshot #10

A blue rectangular object with numbers

Description automatically generated

Figure 19. Screenshot #11

A screen shot of a computer

Description automatically generated

Figure 20. Source code #9

A screenshot of a computer program

Description automatically generated

Figure 21. Source code #10

A screen shot of a computer

Description automatically generated

Figure 22. Source code #11

## **Task #7. Saving all game progress, loading it after closing game**

Using Data Store, all game progress is stored locally, every time any progress value is updated. It is retrieved after opening game again. Saved values are: player’s money, amount of fish caught, date and time of every caught fish, names of owned lakes, fishing location’s lake’s index of list of all possible locations.

A screenshot of a phone

Description automatically generated

Figure 23. Screenshot #12

A computer screen shot of a program

Description automatically generated

Figure 24. Source code #12

A screenshot of a computer program

Description automatically generated

Figure 25. Source code #13

A screen shot of a computer code

Description automatically generated

Figure 26. Source code #14

A screen shot of a computer code

Description automatically generated

Figure 27. Source code #15

## **Task #8. Creating a shop where player can buy new fishing locations**

After pressing the shop button in the menu screen player can see a list of items on sale. Each salable lake has a picture, name of location, its price and buy button if player hasn’t bought it yet. This information is displayed using an adapter in recycle view, making shopping list scrollable.

A screenshot of a computer

Description automatically generated

Figure 28. Screenshot #13

A collage of a lake

Description automatically generated

Figure 29. Screenshot #14

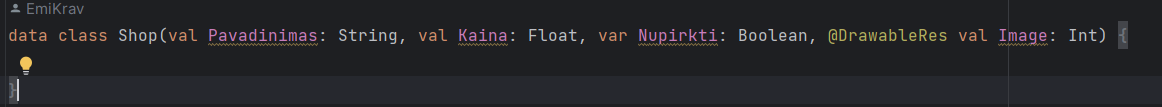


Figure 30. Source code #16

A screenshot of a computer program

Description automatically generated

Figure 31. Source code #17

A computer screen shot of a program

Description automatically generated

Figure 32. Source code #18

A screen shot of a computer program

Description automatically generated

Figure 33. Source code #19

## **Task #9. Changing fishing location from a list of owned location**

While waiting for fish to take the bait, the player can press compass button to go to another location. If the player owns any lakes, background image changes to the next one in line of list of all lakes. After catching or failing to catch a fish, the player is returned to their chosen location.

A fishing pole on a dock

Description automatically generated

Figure 34. Screenshot #15

A fishing pole in the water

Description automatically generated

Figure 35. Screenshot #16

A person in a scuba diving suit in the water

Description automatically generated

Figure 36. Screenshot #17

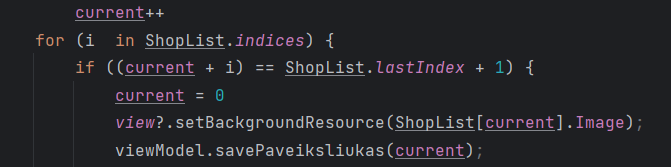


Figure 37. Source code #20

A screenshot of a computer

Description automatically generated

Figure 38. Source code #21

A computer screen shot of a program code

Description automatically generated

Figure 39. Source code #22

## **Task #10. Animating an image**

After starting the game, after a random amount of time (from 3 to 10 seconds), the fishing float starts moving random distance to the left or to the right from its starting position. After every move it moves again in the opposite direction. This continues until the player either presses on a fishing rod or amount of time to catch the fish ends and fishing float returns to its starting position.

A fishing pole in the water

Description automatically generated

Figure 40. Screenshot #18

A group of people fishing in the water

Description automatically generated

Figure 41. Screenshot #19

A screen shot of a computer program

Description automatically generated

Figure 42. Source code #23

A computer screen shot of a program code

Description automatically generated

Figure 44. Source code #24

# **Reference list**

1. Source #1. <https://www.youtube.com/watch?v=TcTgbVudLyQ>
2. Source #2. <https://developer.android.com>
3. Source #3. <https://kotlinlang.org/docs/home.html>
4. <https://www.geeksforgeeks.org/how-to-load-any-image-from-url-without-using-any-dependency-in-android/>
5. <https://fishingbooker.com/fish>