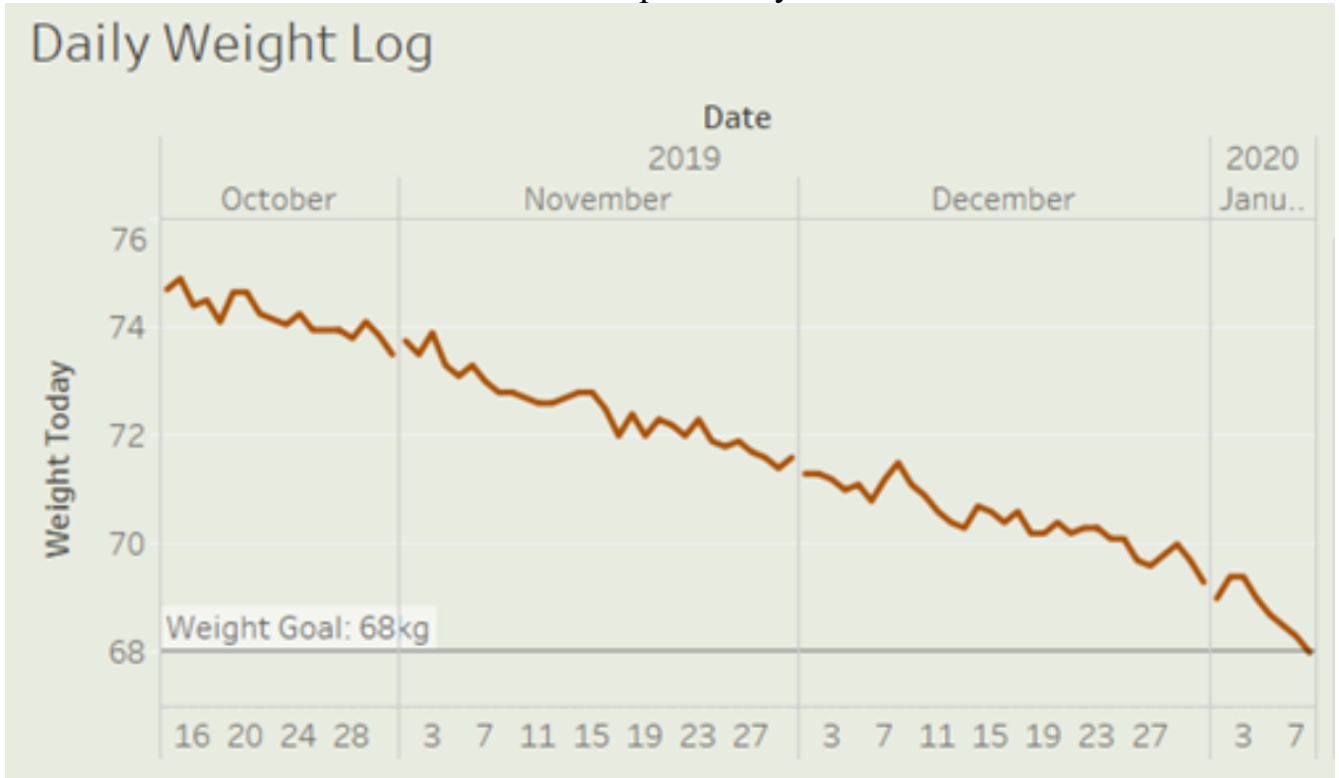


ДОДАТОК А

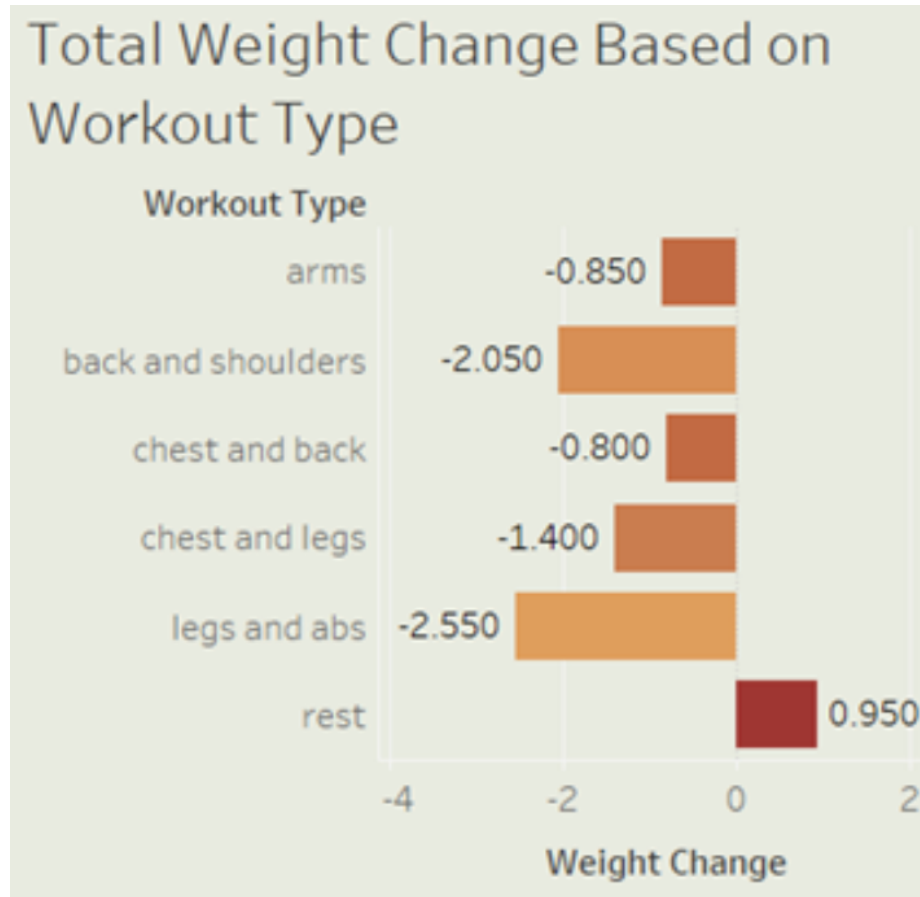
Графічні матеріали

Як виглядає процес схуднення?



Змін.	Арк.	№ докум.	Підп.	Дата	Дослідження та розробка методу прогнозування параметрів статури людини для системи планування тренувань	Аркуш	Аркушів.
Розробив		Петренко А.Ю.				1	1
Перевірив		Калита Н. І.					
Т. Контр.							
Реценз.							
Н.контр.		Калита Н. І.				ХНУРЕ кафедра СТ	
Утверд.		Гребеннік І. В.					

Значущість фізичних навантажень



Змін.	Арк.	№ докум.	Підп.	Дата	Дослідження та розробка методу прогнозування параметрів статури людини для системи планування тренувань	Аркуш	Аркушів.
Розробив		Петренко А.Ю.				1	1
Перевірив		Калита Н. І.					
Т. Контр.							
Реценз.							
Н.контр.		Калита Н. І.				ХНУРЕ кафедра СТ	
Утверд.		Гребеннік І. В.					

Таблиця 2.3 – Початкові дані.

Вага поточна	Вага у наступний день	Залишок калорій	Тип тренування
72.5	72	-56.31	Груди та ноги
72	72.4	-60.41	Спина та плечі
72.4	72	-68.61	Ноги та прес
72	72.2	-40.4	Руки
72.2	72	-37.08	Груди та спина
72	72.3	46.71	Відпочинок

Таблиця 2.4 – Таблиця ефективності тренувань.

Руки	0.33
Спина та плечі	0.804
Груди та спина	0.3137
Груди та ноги	0.549
Ноги та прес	1
Відпочинок	0.3725

Таблиця 2.5 – Порівняння результатів регресій.

Прогнозовані залежні змінні	Регресія з ваговим коефіцієнтом	Звичайна багатофакторна регресія
\hat{y}_1	72,485	72,578
\hat{y}_2	72,05	72,093
\hat{y}_3	72,13	72,13
\hat{y}_4	71,959	72,124
\hat{y}_5	72,122	72,222
\hat{y}_6	71,927	72,011

Змін.	Арк.	№ докум.	Підп.	Дата	Дослідження та розробка методу прогнозування параметрів статури людини для системи планування тренувань	Аркуш	Аркушів.
Розробив		Петренко А.Ю.				1	1
Перевірив		Калита Н. І.					
Т. Контр.							
Реценз.							
Н.контр.		Калита Н. І.				ХНУРЕ кафедра СТ	
Утверд.		Гребеннік І. В.					

ДОДАТОК Б

Текст програми

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

«ЗАТВЕРДЖУЮ»

Керівник кваліфікаційної роботи

_____ доц. Тітов С.В.
(підпис)

ДОСЛІДЖЕННЯ ТА РОЗРОБКА МЕТОДУ ПРОГНОЗУВАННЯ
ПАРАМЕТРІВ СТАТУРИ ЛЮДИНИ ДЛЯ СИСТЕМИ ПЛАНУВАННЯ
ТРЕНУВАНЬ

Текст програми

ЛИСТ ЗАТВЕРДЖЕННЯ

УЗГОДЖЕНО:

РОЗРОБИВ:

ст. гр. ІТПМ-20-1

Петренко А.Ю.

ЗАТВЕРДЖЕНО

ДОСЛІДЖЕННЯ ТА РОЗРОБКА МЕТОДУ ПРОГНОЗУВАННЯ
ПАРАМЕТРІВ СТАТУРИ ЛЮДИНИ ДЛЯ СИСТЕМИ ПЛАНУВАННЯ
ТРЕНУВАНЬ

Текст програми

Аркушів 8

```

// LoginViewController.swift
// ARFitHelper
//
// Created by Антон Петренко on 16.01.2020.
// Copyright © 2020 Антон Петренко. All rights reserved.
//

import UIKit
import Firebase

class LoginViewController: UIViewController {

    @IBOutlet weak var lowLabel: UILabel!

    @IBOutlet weak var loginTextField: UITextField!

    @IBOutlet weak var passwordTextField: UITextField!

    @IBOutlet weak var activityIndicator: UIActivityIndicatorView!

    override func viewDidLoad(_ animated: Bool) {
        super.viewDidLoad(animated)
        lowLabel.text = ""
    }

    override func viewWillAppear(_ animated: Bool) {
        super.viewWillAppear(animated)
        activityIndicator.isHidden = true
    }

    override func viewDidDisappear(_ animated: Bool) {
        super.viewDidDisappear(animated)
        activityIndicator.stopAnimating()
    }

    @IBAction func loginButtonPress(_ sender: Any) {

        activityIndicator.isHidden = false
        activityIndicator.startAnimating()

        if validLogin(loginTextField.text) && validPassword(passwordTextField.text) {

            let login = loginTextField.text!
            let password = passwordTextField.text!

            if(!login.isEmpty && !password.isEmpty) {
                Auth.auth().signIn(withEmail: login, password: password) { [weak self] (result, error) in

                    if error != nil {
                        self?.activityIndicator.stopAnimating()
                        self?.activityIndicator.isHidden = true
                        self?.showAlert()
                    } else if error == nil {
                        UserDefaults.standard.set(true, forKey: "isLoggedIn")
                        let startingPage: UIStoryboard = UIStoryboard(name: "StartPage", bundle: nil)
                        let mainViewController = startingPage.instantiateViewController(withIdentifier: "MainPageID") as! MainPageTabBarController
                        self?.present(mainViewController, animated: true)
                    }
                }
            } else {
                showAlert()
            }
        } else {
            self.activityIndicator.isHidden = true
            self.activityIndicator.stopAnimating()
            lowLabel.text = "Wrong login or password!"
        }
    }

    @IBAction func logOutButtonPress(_ sender: Any) {
        UserDefaults.standard.set(false, forKey: "isLoggedIn")
    }
}

```

```

    let registerPage = UIStoryboard(name: "Register", bundle: Bundle.main)
    let registerViewController = registerPage.instantiateViewController(identifier: "RegisterViewControllerID") as! RegisterViewController
    self.present(registerViewController, animated: true)
}

func showAlert() {
    self.activityIndicator.isHidden = true
    self.activityIndicator.stopAnimating()
    let alert = UIAlertController(title: "Error", message: "Fill all fields with correct data, please!", preferredStyle: .alert)
    alert.addAction(UIAlertAction(title: "Ok", style: .default, handler: nil))
    present(alert, animated: true)
}
}
// RegisterViewController.swift
// ARFitHelper
//
// Created by Антон Петренко on 19.01.2020.
// Copyright © 2020 Антон Петренко. All rights reserved.
//

import UIKit
import Firebase
import FirebaseDatabase

class RegisterViewController: UIViewController {

    @IBOutlet weak var logInOrRegLabel: UILabel!

    @IBOutlet weak var nameTextField: UITextField!

    @IBOutlet weak var surnameTextField: UITextField!

    @IBOutlet weak var emailTextField: UITextField!

    @IBOutlet weak var passwordTextField: UITextField!

    @IBOutlet weak var haveAnAccountLabel: UILabel!

    @IBOutlet weak var registerButton: UIButton!

    override func viewDidLoad() {
        super.viewDidLoad()

        self.registerButton.layer.cornerRadius = 14
    }

    @IBAction func registerButtonPress(_ sender: Any) {

        let name = nameTextField.text!
        let email = emailTextField.text!
        let password = passwordTextField.text!

        if(!name.isEmpty && !email.isEmpty && !password.isEmpty) {
            Auth.auth().createUser(withEmail: email, password: password) { (result, error) in

                if error != nil {
                    let alert = UIAlertController(title: "Error", message: error?.localizedDescription, preferredStyle: .alert)
                    alert.addAction(UIAlertAction(title: "Ok", style: .default, handler: nil))
                    self.present(alert, animated: true)
                } else if error == nil {
                    if let result = result {
                        let ref = Database.database().reference().child("users")

                        ref.child(result.user.uid).updateChildValues(["name" : name, "email" : email, "surname": self.surnameTextField.text!])
                        self.dismiss(animated: true, completion: nil)

                        let alert = UIAlertController(title: "Registration", message: "Registration was completed!", preferredStyle: .alert)
                        alert.addAction(UIAlertAction(title: "Ok", style: .default, handler: { (action) in
                            let startingPage = UIStoryboard(name: "StartPage", bundle: nil)
                            let mainViewController = startingPage.instantiateViewController(withIdentifier: "MainPageID") as!
                                MainPageTabBarController
                            self.present(mainViewController, animated: true)
                        }))
                        self.present(alert, animated: true)
                    }
                }
            }
        }
    }
}

```

```

        }
    }
} else {
    showAlert()
}
}

func showDBorNetworkAlert() {
    let alert = UIAlertController(title: "Error", message: "Network or database error", preferredStyle: .alert)
    alert.addAction(UIAlertAction(title: "Ok", style: .default, handler: nil))
    self.present(alert, animated: true)
}

func showAlert() {
    let alert = UIAlertController(title: "Error", message: "Fill all fields, please!", preferredStyle: .alert)
    alert.addAction(UIAlertAction(title: "Ok", style: .default, handler: nil))
    present(alert, animated: true)
}

@IBAction func backButtonWasPressed(_ sender: Any) {
    let startingPage: UIStoryboard = UIStoryboard(name: "Main", bundle: nil)
    let mainViewController = startingPage.instantiateViewController(withIdentifier: "MainID") as! LoginViewController
    self.present(mainViewController, animated: true)
}
}
// ManualTableViewController.swift
// ARFitHelper
//
// Created by АНТОН Петренко on 21.01.2020.
// Copyright © 2020 АНТОН Петренко. All rights reserved.
//

import UIKit
import Firebase

class ManualTableViewController: UITableViewController, UISearchResultsUpdating {

    var databaseReference: DatabaseReference?
    var databaseHandle: DatabaseHandle?

    var newExc: Exercise?

    var resultSearchController = UISearchController()

    @IBOutlet var myTableView: UITableView!

    // var exercises = [Exercise]()
    var selectedExercises = [Exercise]()

    var filteredTableData = [Exercise]()

    override func viewDidLoad() {
        super.viewDidLoad()
        if oneTime == true {
            getDataFromDatabase()
            oneTime = false
        }
        myTableView.accessibilityIdentifier = "ManualTable"
        myTableView.dataSource = self
        myTableView.delegate = self
        let nib = UINib(nibName: "ManualTableViewCell", bundle: nil)
        myTableView.register(nib, forCellReuseIdentifier: "ManualCellID")

        // MARK: Configure search bar

        resultSearchController = ({
            let controller = UISearchController(searchResultsController: nil)
            controller.searchResultsUpdater = self
            controller.dimsBackgroundDuringPresentation = false
            controller.searchBar.sizeToFit()

            tableView.tableHeaderView = controller.searchBar
        })
    }
}

```

```

        return controller
    })()

    tableView.reloadData()
}

override func viewWillAppear(_ animated: Bool) {
    super.viewWillAppear(animated)

    tableView.reloadData()
}

func getDataFromDatabase() {
    databaseReference = Database.database().reference()

    databaseHandle = databaseReference?.child("exercises").observe(.childAdded, with: { (snapshot) in

        let exerciseFromDB = snapshot.value

        if let dict = exerciseFromDB as? [AnyHashable: Any]{
            let exc = Exercise(name: dict["exercisename"] as? String ?? "",
, kind: dict["kind"] as? String ?? "", targetingMuscles: dict["targetingMusclesGroup"] as? String ?? "", synergistsMuscles:
dict["synergistsMusclesGroup"] as? String ?? "", technique: dict["technic"] as? String ?? "", videoURL: dict["videoURL"] as? String ?? "")

            exercises.append(exc)
        }
        self.myTableView.reloadData()
    })
}

// MARK: - Table view data source

override func tableView(_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {

    if (resultSearchController.isActive) {
        return filteredTableData.count
    } else {
        return exercises.count
    }
}

override func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
    let cell = tableView.dequeueReusableCell(withIdentifier: "ManualCellID", for: indexPath) as! ManualTableViewCell
    cell.accessibilityIdentifier = "manualCell_\(indexPath.row)"

//    cell.backgroundColor = #colorLiteral(red: 0.8727599978, green: 0.8838961124, blue: 0.8835354447, alpha: 1)
//    cell.backgroundView = UIImageView(image: UIImage(named: "lines.png"))
//    cell.backgroundView?.contentMode = .

    if (resultSearchController.isActive) {

        cell.configure(filteredTableData[indexPath.row])
        return cell
    }
    else {
        cell.configure(exercises[indexPath.row])

        return cell
    }
}

override func tableView(_ tableView: UITableView, didSelectRowAt indexPath: IndexPath) {
    resultSearchController.isActive = false
    let page: UIStoryboard = UIStoryboard(name: "DetailedExercise", bundle: Bundle.main)
    let viewController = page.instantiateViewController(withIdentifier: "DetailedManualID") as! DetailedManualViewController
    viewController.exc = exercises[indexPath.row]
//    self.present(viewController, animated: true)
    resultSearchController.isEditing = false
    self.navigationController?.pushViewController(viewController, animated: true)
    self.tableView.deselectRow(at: indexPath, animated: true)
}

```

```

@IBAction func backButtonPress(_ sender: Any) {
    let storyboard = UIStoryboard(name: "StartPage", bundle: Bundle.main)
    let viewController = storyboard.instantiateViewController(withIdentifier: "MainPageID")
    self.present(viewController, animated: true)
}

func updateSearchResults(for searchController: UISearchController) {
    filteredTableData.removeAll(keepingCapacity: false)

    let searchPredicate = NSPredicate(format: "SELF CONTAINS[c] %@", searchController.searchBar.text!)

    let res = exercises.filter{ searchPredicate.evaluate(with: $0.name) || searchPredicate.evaluate(with: $0.kind) || searchPredicate.evaluate(with: $0.targetingMuscles)}

    filteredTableData = res

    self.tableView.reloadData()
}

}

// DetailedManualViewController.swift
// ARFitHelper
//
// Created by Антон Петренко on 23.01.2020.
// Copyright © 2020 Антон Петренко. All rights reserved.
//

import UIKit
import WebKit

class DetailedManualViewController: UIViewController, WKNavigationDelegate {

    var exc: Exercise?

    @IBOutlet weak var nameLabel: UILabel!
    @IBOutlet weak var kindLabel: UILabel!
    @IBOutlet weak var targetingLabel: UILabel!
    @IBOutlet weak var synergistLabel: UILabel!
    @IBOutlet weak var technicLabel: UILabel!
    @IBOutlet weak var webKit: WKWebView!
    @IBOutlet weak var activityLoader: UIActivityIndicatorView!

    override func viewDidLoad() {
        super.viewDidLoad()

        self.webKit.navigationDelegate = self
        self.webKit.isAccessibilityElement = true
        self.webKit.accessibilityIdentifier = "webViewVideo"

        title = "Exercise"

        nameLabel.text = "Name: " + exc!.name
        kindLabel.text = "Type: " + exc!.kind
        targetingLabel.text = "Targeting: " + exc!.targetingMuscles
        synergistLabel.text = "Synergists: " + exc!.synergistsMuscles
        technicLabel.text = "Technic: " + exc!.technic

        getVideo(URL(string: exc!.videoURL)!)
    }

    func webView(_ webView: WKWebView, didStartProvisionalNavigation navigation: WKNavigation!) {

```

```

        activityLoader.startAnimating()
        activityLoader.isHidden = false
    }

    func webView(_ webView: WKWebView, didFinish navigation: WKNavigation!) {
        activityLoader.stopAnimating()
        activityLoader.isHidden = true
    }

// override func observeValue(forKeyPath keyPath: String?, of object: Any?, change: [NSKeyValueChangeKey : Any]?, context:
UnsafeMutableRawPointer?) {
//     if keyPath == "loading" {
//         if webView.isLoading {
//             activityLoader.startAnimating()
//             activityLoader.isHidden = false
//         } else {
//             activityLoader.stopAnimating()
//         }
//     }
// }

@IBAction func backButtonPress(_ sender: Any) {
    let storyboard = UIStoryboard(name: "StartPage", bundle: Bundle.main)
    let viewController = storyboard.instantiateViewController(withIdentifier: "MainPageID")
    self.present(viewController, animated: true)
}

func getVideo(_ videoURL: URL) {
    webView.load(URLRequest(url: videoURL))
}

func create(_ exercise: Exercise) {
    self.exc = exercise
}

// func webViewDidStartLoad(_ webView: UIWebView) {
//     activityLoader.isHidden = false
//     activityLoader.startAnimating()
// }
//
// func webViewDidFinishLoad(_ webView: UIWebView) {
//     activityLoader.stopAnimating()
//     activityLoader.isHidden = true
// }
// AddingExerciseViewController.swift
// ARFitHelper
//
// Created by Антон Петренко on 29.01.2020.
// Copyright © 2020 Антон Петренко. All rights reserved.
//

import UIKit

class AddingExerciseViewController: UIViewController {

    @IBOutlet weak var nameLabel: UILabel!

    @IBOutlet weak var kindLabel: UILabel!

    @IBOutlet weak var targetingLabel: UILabel!

    @IBOutlet weak var synergistsLabel: UILabel!

    @IBOutlet weak var technicLabel: UILabel!

    @IBOutlet weak var videourlLabel: UILabel!

    @IBOutlet weak var nameTextField: UITextField!

    @IBOutlet weak var kindTextField: UITextField!

    @IBOutlet weak var targetingMusclesTextField: UITextField!

    @IBOutlet weak var synergistMusclesTextField: UITextField!

```

```

@IBOutlet weak var technicTextField: UITextField!

@IBOutlet weak var videoURLTextField: UITextField!

@IBOutlet weak var addExerciseButton: UIButton!

override func viewDidLoad() {
    super.viewDidLoad()
    self.view.backgroundColor = UIColor(patternImage: UIImage(named: "pumpit7.png")!)
    self.nameTextField.accessibilityIdentifier = "NewExerciseName"
    self.kindTextField.accessibilityIdentifier = "NewExerciseKind"
    self.targetingMusclesTextField.accessibilityIdentifier = "NewExerciseTargetingMuscles"
    self.synergistMusclesTextField.accessibilityIdentifier = "NewExerciseSynergistsMuscles"
    self.technicTextField.accessibilityIdentifier = "NewExerciseTechnic"
    self.videoURLTextField.accessibilityIdentifier = "NewExerciseVideoURL"
    configureButton()
    setStandardLabelsTexts()
}

@IBAction func addButtonPress(_ sender: UIButton) {
    if fillingCheckForTextField(nameTextField.text ?? "") && fillingCheckForTextField(kindTextField.text ?? "") &&
fillingCheckForTextField(targetingMusclesTextField.text ?? "") && fillingCheckForTextField(synergistMusclesTextField.text ?? "") &&
fillingCheckForTextField(technicTextField.text ?? "") && fillingCheckForTextField(videoURLTextField.text ?? "") {
        exercises.append(Exercise(name: nameTextField.text!, kind: kindTextField.text!, targetingMuscles: targetingMusclesTextField.text!,
synergistsMuscles: synergistMusclesTextField.text!, technic: technicTextField.text!, videoURL: videoURLTextField.text!))
        let alert = UIAlertController(title: "Adding exercise", message: "New exercise was added! Thank you!", preferredStyle: .alert)
        let okAction = UIAlertAction(title: "Ok", style: .default, handler: nil)
        alert.addAction(okAction)
        present(alert, animated: true, completion: nil)
    } else {
        let alert = UIAlertController(title: "Adding exercise", message: "Can't add exercise! It doesn't conform to minimal amount of chars!",
preferredStyle: .alert)
        let okAction = UIAlertAction(title: "Ok", style: .default, handler: nil)
        alert.addAction(okAction)
        present(alert, animated: true, completion: nil)
    }
}

func fillingCheckForTextField(_ text: String) -> Bool {
    if text.count > 5 {
        return true
    } else {
        return false
    }
}

func configureButton() {
    addExerciseButton.backgroundColor = .black
}

func setStandardLabelsTexts() {
    nameLabel.text = "Name:"
    kindLabel.text = "Kind:"
    targetingLabel.text = "Targetind muscles:"
    synergistsLabel.text = "Synergists muscles:"
    technicLabel.text = "Technic:"
    videourlLabel.text = "Video URL:"
}

/*
// MARK: - Navigation

// In a storyboard-based application, you will often want to do a little preparation before navigation
override func prepare(for segue: UIStoryboardSegue, sender: Any?) {
    // Get the new view controller using segue.destination.
    // Pass the selected object to the new view controller.
}
*/
}
// ExerciseSetViewController.swift
// ARFitHelper
//

```

```

// Created by Антон Петренко on 29.01.2020.
// Copyright © 2020 Антон Петренко. All rights reserved.
//

var totalnum: Int = 0

import UIKit

class ExerciseSetViewController: UIViewController {

    @IBOutlet weak var nextButton: UIButton!

    var nums = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

    @IBOutlet weak var pickerView: UIPickerView!

    override func viewDidLoad() {
        super.viewDidLoad()
        self.view.backgroundColor = UIColor(patternImage: UIImage(named: "sportgirl2.png")!)
        pickerView.delegate = self
        pickerView.dataSource = self
        configureButton()
        // Do any additional setup after loading the view.
    }

    @IBAction func nextButtonTap(_ sender: Any) {
        let startingPage: UIStoryboard = UIStoryboard(name: "CurrentExercise", bundle: Bundle.main)
        let mainViewController = startingPage.instantiateViewController(withIdentifier: "CurrExcID") as! CurrentExerciseViewController
        self.present(mainViewController, animated: true)
    }

    func configureButton() {
        nextButton.layer.borderWidth = 1
        nextButton.layer.borderColor = UIColor.black.cgColor
        nextButton.layer.cornerRadius = 10
    }
}

extension ExerciseSetViewController: UIPickerViewDelegate, UIPickerViewDataSource {

    func numberOfComponents(in pickerView: UIPickerView) -> Int {
        return 1
    }

    func pickerView(_ pickerView: UIPickerView, numberOfRowsInComponent component: Int) -> Int {
        return nums.count
    }

    func pickerView(_ pickerView: UIPickerView, titleForRow row: Int, forComponent component: Int) -> String? {
        return String(nums[row])
    }

    func pickerView(_ pickerView: UIPickerView, didSelectRow row: Int, inComponent component: Int) {
        totalnum = nums[row]
        print(totalnum)
    }
}

```

ДОДАТОК В

Відомість кваліфікаційної роботи

