МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

**«Национальный исследовательский**

**Нижегородский государственный университет им. Н.И. Лобачевского»**

**(ННГУ)**

**Институт информационных технологий, математики и механики**

Направление подготовки: «Программная инженерия»

**ОТЧЕТ**

по лабораторной работе №3

на тему

**««Типовые элементы интерфейса на мобильных устройствах»**

**Выполнил:** студент группы 381808-2 Евсеев А.Д.

**Преподаватель:**

Карчков Д.А.

**Нижний Новгород**

**2021**

**Цель работы**

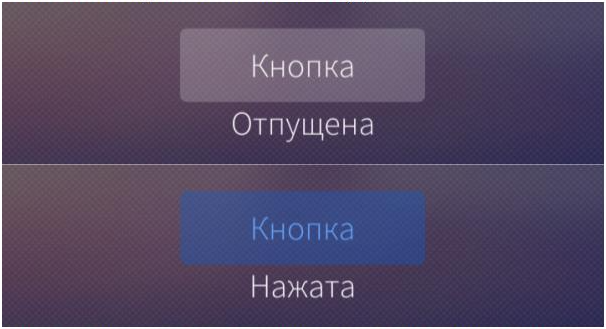
Научиться применять типовые элементы интерфейса Sailfish OS.

**Шаги**:

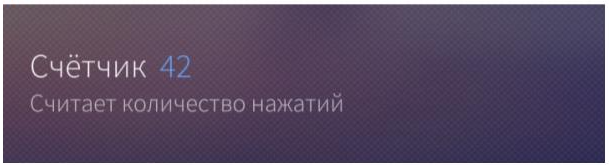
1) создать текстовое поле для ввода числа с заголовком и подсказкой

2) создать кнопку, которая будет сохранять визуально нажатое состояние, после того, как пользователь нажал на неё один раз

3) создать кнопку и поле с текстом. Поле с текстом должно отображать нажата ли кнопка или нет выводом текста “Нажата” или “Отпущена”



4) создать кнопку со значением, которая будет отображать количество нажатий на нее



5) создать селектор даты, который будет отображать выбранную дату в консоли

6) создать селектор времени, который будет отображать выбранное время в консоли

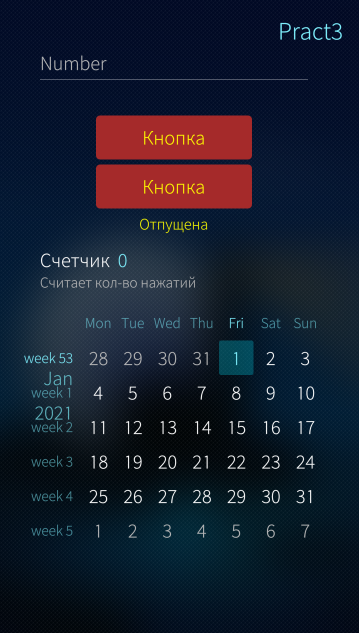
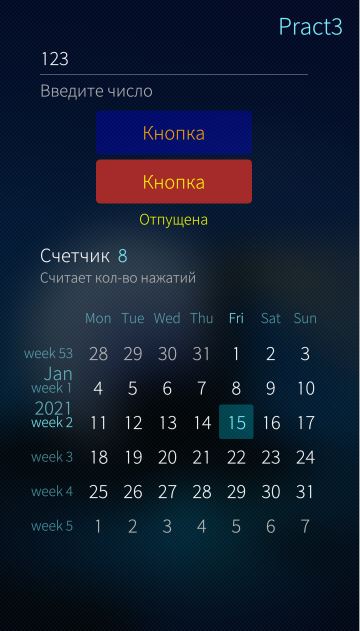
7) создать поле с выпадающим списком, позволяющее выбрать строку из списка. Результат выбора отобразить в консоли

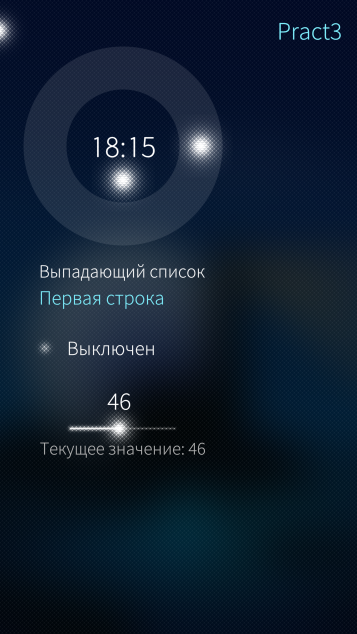
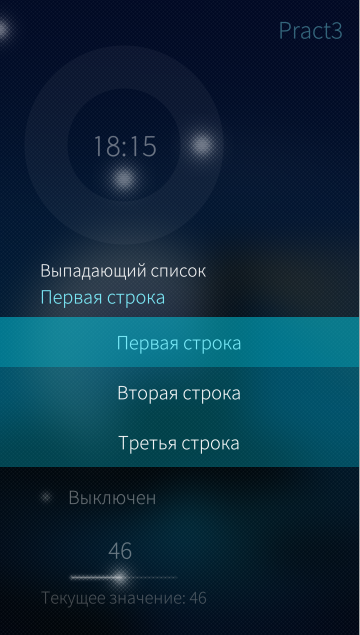
8) создать переключатель с текстом, в тексте отобразить состояние переключателя “Включен” или “Выключен”

9) создать ползунок и поле с текстом. Поле с текстом должно отображать текущее значение ползунка



**Демонстрация работы**

** **

** **

**Заключение**

В ходе практической работы было реализовано приложение, содержащее все описанные шаги. Был рассмотрен TextField и его основные свойства. Подробно познакомились с элементами управления: Button, ValueButton, DatePicker, TimePicker, ComboBox, TextSwitch и Slider.

**Приложения**

1. Код FirstPage.qml:

import QtQuick 2.0

import Sailfish.Silica 1.0

import QtQuick 2.0

import Sailfish.Silica 1.0

Page {

id: *page*

allowedOrientations: Orientation.All

SilicaFlickable {

anchors.fill: *parent*

PullDownMenu {

MenuItem {

text: *qsTr*("Show Page 2")

onClicked: *pageStack*.push(*Qt*.resolvedUrl("SecondPage.qml"))

}

}

contentHeight: *column*.height

Column {

id: *column*

width: *page*.width

spacing: Theme.paddingLarge

PageHeader {

title: *qsTr*("Pract3")

}

Item {

x: 50

y: 100

Column {

spacing: 10

TextField {

label: "Введите число"

placeholderText: "Number"

}

Button {

anchors.horizontalCenter: *parent*.horizontalCenter

color: "yellow"

backgroundColor: "brown"

highlightColor: "orange"

highlightBackgroundColor: "blue"

text: "Кнопка"

onClicked: *down* = !*down*

}

Button {

id: *but2*

anchors.horizontalCenter: *parent*.horizontalCenter

color: "yellow"

backgroundColor: "brown"

highlightColor: "orange"

highlightBackgroundColor: "blue"

text: "Кнопка"

}

Text {

id: *text*

anchors.horizontalCenter: *parent*.horizontalCenter

text: *but2*.down == true ? "Нажата" : "Отпущена";

color: "yellow"

}

ValueButton {

label: "Счетчик"

description: "Считает кол-во нажатий"

value: "0"

onClicked: *value* = parseInt(*value*) + 1

}

DatePicker {

width: 600

height: 600

date: new *Date*(2021, 0, 1, 0, 0, 0)

daysVisible: true

weeksVisible: true

monthYearVisible: true

onDateTextChanged: *console*.log(*dateText*)

}

}

}

}

}

}

1. Код SecondPage.qml:

import QtQuick 2.0

import Sailfish.Silica 1.0

Page {

id: *page*

allowedOrientations: Orientation.All

SilicaFlickable {

anchors.fill: *parent*

PullDownMenu {

MenuItem {

text: *qsTr*("Show Page 1")

onClicked: pageStack.push(*Qt*.resolvedUrl("FirstPage.qml"))

}

}

contentHeight: *column*.height

Column {

id: *column*

width: *page*.width

spacing: Theme.paddingLarge

PageHeader {

title: *qsTr*("Pract3")

}

Item {

x: 50

y: 100

Column {

spacing: 10

TimePicker {

width: 400

height: 400

id: *timePicker*

hour: 12

minute: 0

Label {

anchors.centerIn: *timePicker*

text: *timePicker*.timeText

font.pixelSize: Theme.fontSizeExtraLarge

}

onTimeTextChanged: *console*.log(*timeText*)

}

ComboBox {

label: "Выпадающий список"

menu: ContextMenu {

MenuItem { text: "Первая строка" }

MenuItem { text: "Вторая строка" }

MenuItem { text: "Третья строка" }

}

onCurrentIndexChanged: *console*.log(*value*)

}

TextSwitch {

text: *checked* ? "Включен" : "Выключен"

}

Slider {

width: *parent*.width

label: "Текущее значение: " + *value*;

minimumValue: 0

maximumValue: 100

value: 0

stepSize: 1

valueText: *value*

}

}

}

}

}