МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ 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»

**Лабораторная работа №3**

По курсу

**«Программирование для мобильных систем»**

**Выполнил(а):** студент группы 381808-2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Алехин Денис

Подпись Андреевич

**Преподаватель:** доцент кафедры ПИ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Борисов Николай

Подпись Анатольевич

Нижний Новгород  
2021

Содержание

[Постановка задачи 3](#_Toc66344188)

[Программная реализация 4](#_Toc66344189)

[Заключение 5](#_Toc66344190)

[Приложения 6](#_Toc66344191)

# Постановка задачи

Научиться применять типовые элементы интерфейса Sailfish OS

# Программная реализация

Для решения поставленной задачи на первую страницу приложения были добавлены элементы TextField, Button, Label, ComboBox, Switch и Slider.

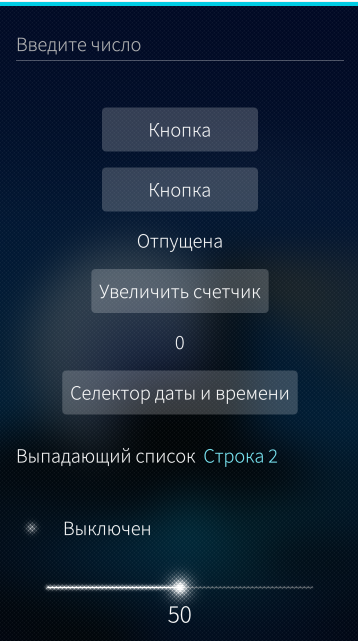
Для изменения значения Label при нажатии на кнопку были использованы события onPressed, onReleased и onClicked.

Так же, для отображение значения элементов ComboBox, Switch и Slider, были использованы комбинации из этих элементов и элемента Label, в которое выводилось значение. При возникновении события onCurrentIdexChanged, onClicked и onValueChanged соответственно, для перечисленных выше элементов, менялось значение элемента Label.

# Заключение

В результате выполнения лабораторной работы были я научился применять типовые элементы интерфейса Sailfish OS

# Приложения

** **

Содержимое файла FirstPage.qml:

import QtQuick 2.0

import Sailfish.Silica 1.0

Page {

id: *page*

// The effective value will be restricted by ApplicationWindow.allowedOrientations

allowedOrientations: Orientation.All

// To enable PullDownMenu, place our content in a SilicaFlickable

SilicaFlickable {

anchors.fill: *parent*

// PullDownMenu and PushUpMenu must be declared in SilicaFlickable, SilicaListView or SilicaGridView

PullDownMenu {

MenuItem {

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

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

}

}

// Tell SilicaFlickable the height of its content.

contentHeight: *column*.height

// Place our content in a Column. The PageHeader is always placed at the top

// of the page, followed by our content.

Column {

id: *column*

y: 50

width: *page*.width

spacing: Theme.paddingLarge

TextField{

width: *parent*.width

inputMethodHints: *Qt*.ImhFormattedNumbersOnly

placeholderText: "Введите число"

}

Button {

text: "Кнопка"

anchors.horizontalCenter: *parent*.horizontalCenter

onClicked: *down* = true

}

Button {

text: "Кнопка"

anchors.horizontalCenter: *parent*.horizontalCenter

onPressed: *label1*.status = "Нажата"

onReleased: *label1*.status = "Отпущена"

}

Label {

id: *label1*

anchors.horizontalCenter: *parent*.horizontalCenter

property string status: "Отпущена"

text: *status*

}

Button {

text: "Увеличить счетчик"

anchors.horizontalCenter: *parent*.horizontalCenter

onClicked: *label2*.counter++

}

Label {

id: *label2*

anchors.horizontalCenter: *parent*.horizontalCenter

property int counter: 0

text: *counter*

}

Button {

text: "Селектор даты и времени"

anchors.horizontalCenter: *parent*.horizontalCenter

onClicked: *pageStack*.push(*dateAndTimePicker*)

}

ComboBox {

id: *comboBox1*

anchors.horizontalCenter: *parent*.horizontalCenter

label: "Выпадающий список"

menu: ContextMenu {

MenuItem { text: "Строка 1"}

MenuItem { text: "Строка 2"}

MenuItem { text: "Строка 3"}

}

onCurrentIndexChanged: *console*.log(*comboBox1*.value)

}

Item {

width: *parent*.width

height: 100

Switch {

id: *switch1*

onClicked:

{

if(*checked* == true) *label3*.state = "Включен"

else *label3*.state = "Выключен"

}

}

Label {

id: *label3*

anchors.verticalCenter: *switch1*.verticalCenter

anchors.left: *switch1*.right

property string state: "Выключен"

text: *state*

}

}

Item {

width: *parent*.width

height: 150

Slider {

id: *slider1*

width: *parent*.width

maximumValue: 100

minimumValue: 0

value: 50

stepSize: 1

onValueChanged: *label4*.value = *value*

}

Label {

id: *label4*

anchors.top: *slider1*.bottom

anchors.horizontalCenter: *parent*.horizontalCenter

property int value: 50

text: *value*

font.pixelSize: Theme.fontSizeLarge

transform: Translate {y: -30 }

}

}

}

}

Dialog {

id: *dateAndTimePicker*

Column {

width:*parent*.width

spacing: 30

DialogHeader {

title: "Выберете дату и время"

}

DatePicker {

cellHeight: 70

id: *date*

daysVisible: true

}

Item {

height: *time*.height

width: *parent*.width

TimePicker {

anchors.horizontalCenter: *parent*.horizontalCenter

width: 400

height: 400

id: *time*

hourMode: DateTime.TwentyFourHours

}

Label {

anchors.centerIn: *time*

text: *time*.timeText

font.pixelSize: Theme.fontSizeExtraLarge

}

}

}

onAccepted: *console*.log(*time*.timeText + ", " + *date*.dateText)

}

}