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**

**Кафедра программной инженерии**

Направление подготовки:

«Фундаментальная информатика и информационные технологии»

**ОТЧЕТ**

по Практическому заданию №4:

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

**Выполнила:** студентка группы 381606-2

Колосова Евгения Александровна

**Научный руководитель:** Доцент кафедры программной инженерии

Борисов Николай Анатольевич

Нижний Новгород  
2019

Введение

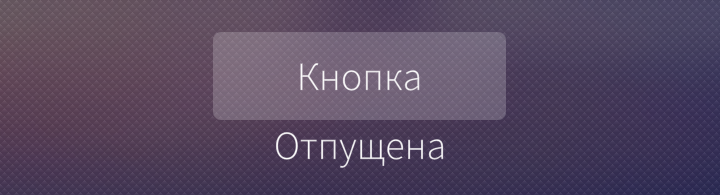
В этой лабораторной работе поднята задача по освоению работы с типовыми элементами управления на мобильных устройствах в Sailfish OS.

Целью работы является приобретение основных навыков взаимодействия с этими элементами, внедрении их в понятный пользователю интерфейс и корректность работы итоговой программы. Следует выполнить все этапы из постановки задачи этой лабораторной работы.

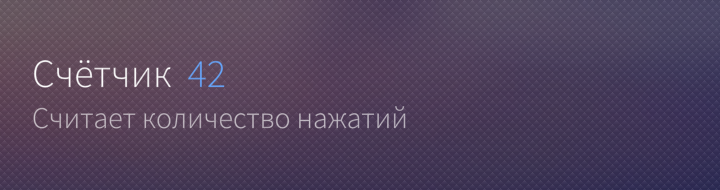
Постановка задачи

Для достижения в результате выполнения работы положительного результата, проект следует реализовывать в соответствии со следующими пунктами:

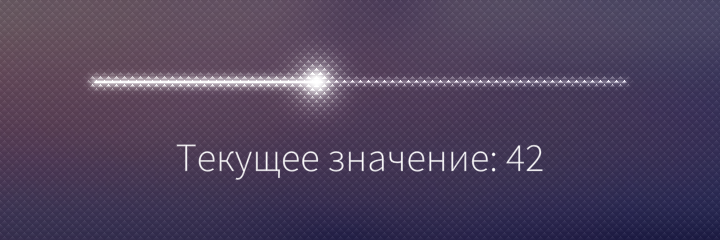
1. Создать текстовое поле для ввода числа с заголовком и подсказкой.
2. Создать кнопку, которая будет сохранять визуально нажатое состояние после того, как пользователь нажал на неё один раз.
3. Создать кнопку и поле с текстом. Поле с текстом должно отображать нажата ли кнопка или нет выводом текста “Нажата” или “Отпущена”.



1. Создать кнопку со значением, которая будет отображать количество нажатий на неё.

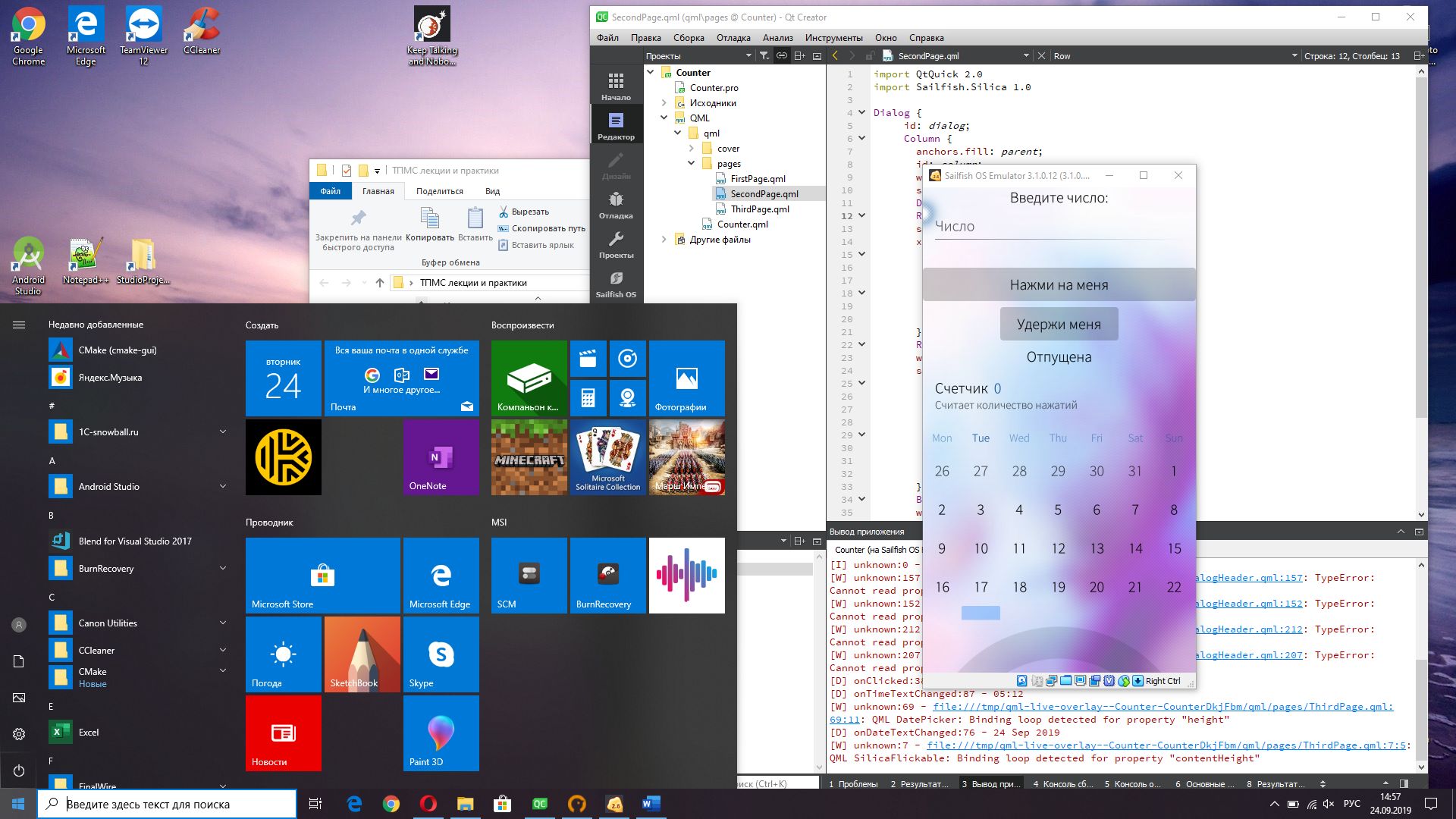


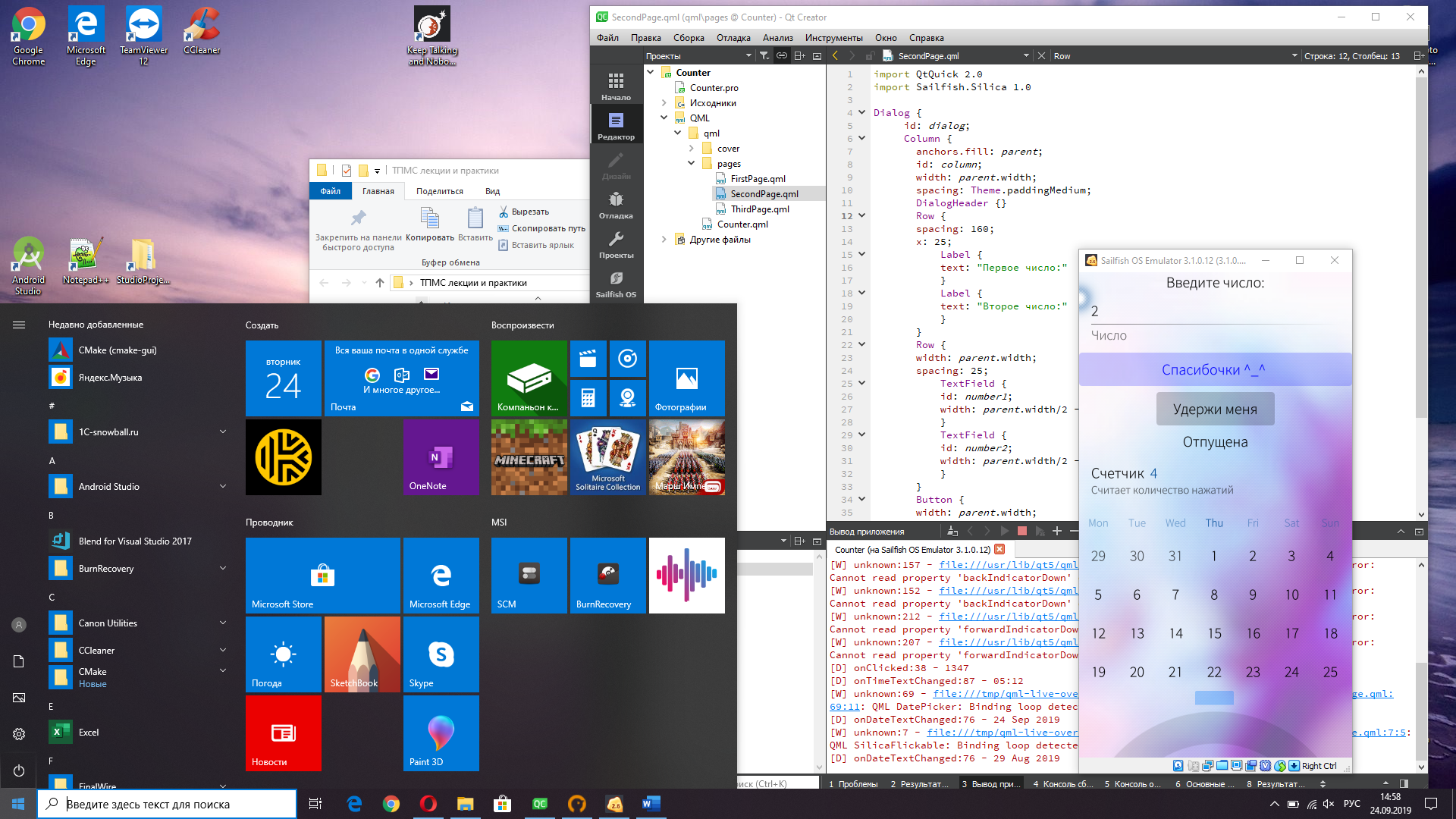
1. Создать селектор даты, который будет отображать выбранную дату в консоли.
2. Создать селектор времени, который будет отображать выбранное время в консоли.
3. Создать поле с выпадающим списком, позволяющее выбрать строку из списка. Результат выбора отобразить в консоли.
4. Создать переключатель с текстом, в тексте отобразить состояние переключателя “Включен” или “Выключен”.
5. Создать ползунок и поле с текстом. Поле с текстом должно отображать текущее значение ползунка.

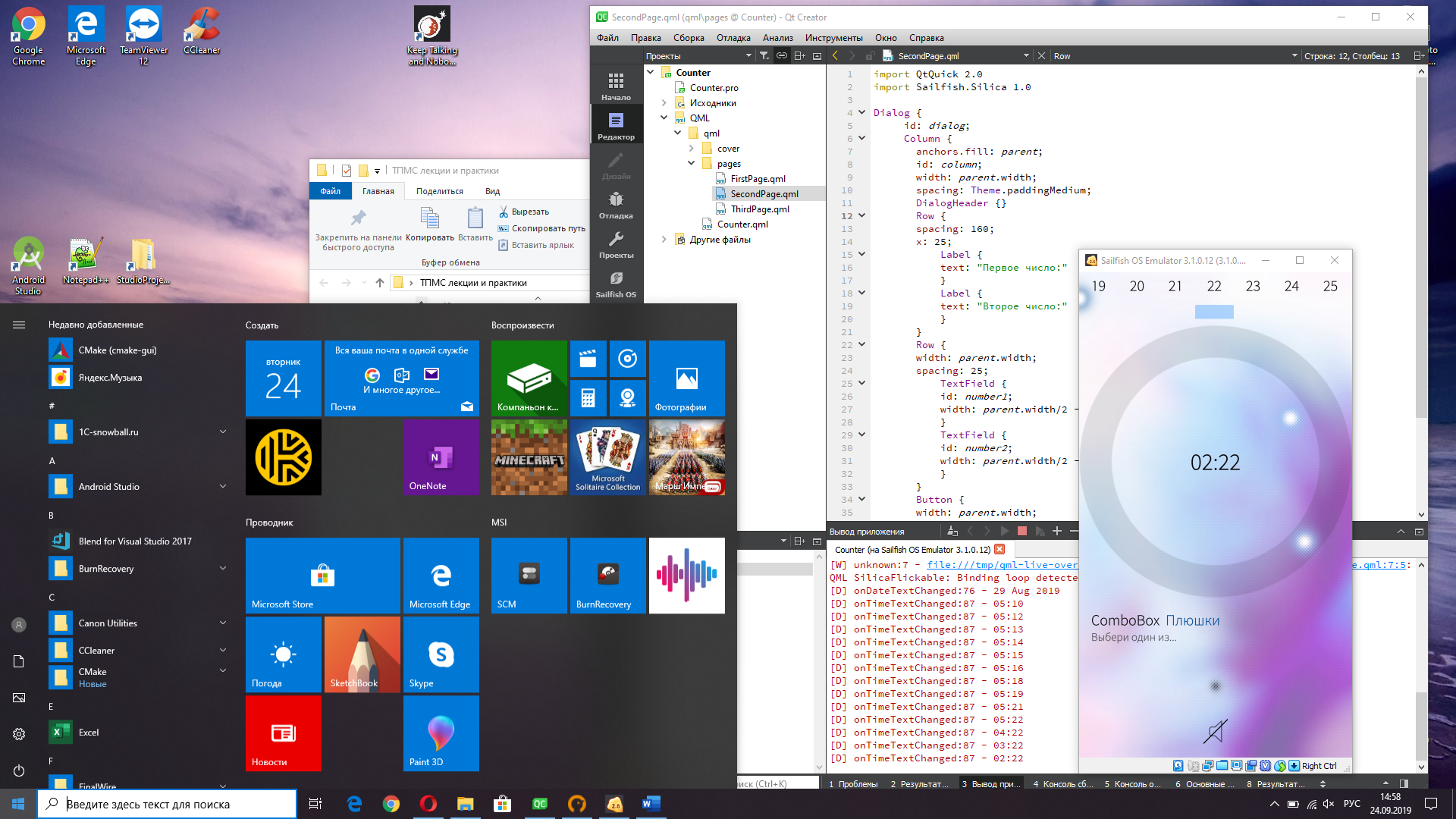


Приложение

Внешний вид приложения и поэтапные вехи его работы:











В итоге на приведенных скринах эмулятора отчетливо видно, что в проекте реализовано и работоспособно все, что было заявлено в постановке задаче этой лабораторной работы. А значит полностью освоен навык работы с типовыми элементами управления мобильных приложений в Sailfish OS.

Код проекта:

FirstPage.qml

1. import QtQuick 2.0
2. import Sailfish.Silica 1.0
4. Page {
5. **id: page2**
6. allowedOrientations: Orientation.All
7. SilicaFlickable {
8. anchors.fill: parent
9. contentHeight: column.height + 1300
11. Column {
12. anchors.fill: parent;
13. id: column;
14. width: parent.width;
15. **spacing: Theme.paddingMedium;**
17. Label {
18. id: labelForNumber;
19. anchors.horizontalCenter: parent.horizontalCenter;
20. **text: "Введите число:"**
21. }
23. TextField {
24. id: textfieldForNumber;
25. **width: parent.width;**
26. label: "Число"
27. placeholderText: label;
28. }
30. **Button {**
31. id:pressedButton;
32. anchors.horizontalCenter: parent.horizontalCenter;
33. width: parent.width;
34. text: "Нажми на меня";
35. **onClicked: {**
36. pressedButton.color = "blue";
37. pressedButton.text = "Спасибочки ^\_^ ";
38. }
39. }
40. **Button {**
41. id: holdedButton;
42. anchors.horizontalCenter: parent.horizontalCenter;
43. highlightBackgroundColor: "lightblue"
44. text: "Удержи меня"
45. **onPressedButtonsChanged: {**
46. **if**(down){
47. holdLabel.text = "Нажата"
48. }
49. **else** {
50. **holdLabel.text = "Отпущена"**
51. }
52. }
53. }
54. Label {
55. **id: holdLabel;**
56. anchors.horizontalCenter: parent.horizontalCenter;
57. text: "Отпущена"
58. }
60. **ValueButton {**
61. width: parent.width;
62. property int count: 0
63. label: "Счетчик"
64. description: "Считает количество нажатий"
65. **value: count;**
66. onClicked: count++;
67. }
69. DatePicker {
70. **id: datePicker;**
71. height: datePicker.height + 500;
72. date: **new** Date();
73. daysVisible: **true**
74. monthYearVisible: **true**
75. **onDateTextChanged:**
76. console.log(dateText);
77. }
79. Item {
80. **height: timePicker.height;**
81. width: parent.width;
82. TimePicker {
83. width: parent.width
84. id: timePicker;
85. **hour: 5**
86. minute: 12
87. onTimeTextChanged: console.log(timeText)
88. }
89. Label {
90. **anchors.centerIn: timePicker;**
91. text: timePicker.timeText
92. font.pixelSize: Theme.fontSizeExtraLarge
93. }
94. }
96. ComboBox {
97. anchors.horizontalCenter: parent.horizontalCenter;
98. label: "ComboBox"
99. description: "Выбери один из..."
100. **menu: ContextMenu {**
101. id: menu
102. MenuItem { text: "Плюшки" }
103. MenuItem { text: "Ватрушки" }
104. MenuItem { text: "Завитушки" }
105. **}**
106. onCurrentIndexChanged: console.log(menu.children[currentIndex].text)
107. }
109. Switch {
110. **anchors.horizontalCenter: parent.horizontalCenter;**
111. id: mute;
112. iconSource: "image://theme/icon-m-speaker-mute?"
113. + (checked ? Theme.highlightColor : Theme.primaryColor);
114. }
115. **Label {**
116. anchors.horizontalCenter: parent.horizontalCenter
117. text: mute.checked?"Включен":"Выключен";
118. }
120. **Slider {**
121. width: parent.width
122. label: "Текущее значение = " + value;
123. minimumValue: 0;
124. maximumValue: 100;
125. **value: 0;**
126. stepSize: 1;
127. }
129. Rectangle {
130. **color: "transparent"**
131. width: parent.width;
132. height: 200;
133. }
134. }
135. **}**
136. }