

РАЗРАБОТКА QML- КОМПОНЕНТА ВИРТУАЛЬНОЙ КЛАВИАТУРЫ ДЛЯ МОБИЛЬНОЙ ПЛАТФОРМЫ АВРОРА

Студентки: Лукьянова Анастасия, Драбкина
Марина
Группа ИС-2301
СПбГЭУ, кафедра информационных
систем и технологий

ТЕМА РАБОТЫ:

РАЗРАБОТКА QML-КОМПОНЕНТА ВИРТУАЛЬНОЙ КЛАВИАТУРЫ ДЛЯ МОБИЛЬНОЙ ПЛАТФОРМЫ АВРОРА.

ПЛАТФОРМА:

ОС АВРОРА (РОССИЙСКИЙ ФОРК SAILFISH OS).

ЧТО СДЕЛАНО:

СОЗДАН ПЕРЕИСПОЛЬЗУЕМЫЙ КОМПОНЕНТ НА QML, КОТОРЫЙ ОБЕСПЕЧИВАЕТ ПОЛНЫЙ КОНТРОЛЬ НАД ВВОДОМ ТЕКСТА, РАБОТАЕТ ПОВЕРХ ИНТЕРФЕЙСА БЕЗ СДВИГА И ПОЗВОЛЯЕТ ПОДКЛЮЧАТЬ СОБСТВЕННЫЙ ОБРАБОТЧИК НАЖАТИЙ С КООРДИНАТАМИ.

ЦЕЛИ И ЗАДАЧИ РАБОТЫ

Цель:

Разработать QML-компонент виртуальной клавиатуры для Авроры, удовлетворяющий пяти ключевым требованиям.

Задачи (основные):

1. Проанализировать существующие решения.
2. Обосновать выбор QML + Sailfish Silica.
3. Спроектировать архитектуру компонента.
4. Реализовать раскладки (рус/англ, символы, Caps Lock).
5. Обеспечить ввод текста в поля TextField/TextArea.
6. Добавить публичный API: `setActiveItem()` и `setKeyHandler()`.
7. Реализовать уникальные ID кнопок и передачу координат нажатий.
8. Провести тестирование и оформить отчёт.

АКТУАЛЬНОСТЬ ПРОБЛЕМЫ

Проблема системной клавиатуры Авроры:

- ▶ Сдвигает интерфейс вверх при появлении.
- ▶ Не даёт перехватывать нажатия и координаты.
- ▶ Не позволяет подключать свой обработчик.

Где нужен более гибкий ввод?

- ▶ Киоск-терминалы.
- ▶ Обучающие приложения.
- ▶ Средства для людей с ограниченными возможностями.
- ▶ Сбор статистики нажатий (лог с координатами).

ХОД РАБОТЫ: АРХИТЕКТУРА И КЛЮЧЕВЫЕ РЕШЕНИЯ

Основные тезисы:

Компонент – один файл Keyboard.qml (легко встроить в любой проект).

Клавиатура расположена поверх UI без сдвига (z:100, отсутствие bottomMargin).

Генерация кнопок через Repeater – минимум кода, легко менять раскладки.

Раскладки хранятся в виде массивов строк (ruLower, ruUpper, enLower, enUpper, symbols).

Переключение языка, регистра, режима символов – через функции toggleLang(), toggleCaps(), toggleMode() + updateKeys().

ИНТЕГРАЦИЯ С ПОЛЯМИ ВВОДА

Как работает ввод:

Поля ввода (TextField/TextArea) при получении фокуса вызывают `keyboard.setActiveItem(this)`.

Вставка символа с учётом курсора:

`insertIntoField(ch)` – разрезает текст и вставляет символ.

Удаление: `deleteFromField()` – удаляет символ перед курсором.

Поддержка пробела, Enter (перевод строки), Backspace.

ПОДКЛЮЧАЕМЫЙ ОБРАБОТЧИК И УНИКАЛЬНЫЕ ID

Главная инновация – внешний обработчик:

Публичный метод `setKeyHandler(handler)`.

На каждое нажатие передаются:

- ▶ `keyId` – уникальный строковый идентификатор (например, `ru_letter_й`, `space`).
- ▶ `keyChar` – символ.
- ▶ `mouseX`, `mouseY` – координаты относительно клавиатуры.
- ▶ `globalX`, `globalY` – глобальные координаты.

Зачем это нужно?

- ▶ Логирование нажатий с точными координатами.
- ▶ Сбор статистики.
- ▶ Специальные реакции на определённые кнопки

ТЕСТИРОВАНИЕ И РЕЗУЛЬТАТЫ

Что тестировали (12 сценариев):

- ▶ Отображение русской/английской раскладок, символов.
- ▶ Переключение языка, Caps Lock, режима 123/ABC.
- ▶ Ввод, удаление, пробел, Enter.
- ▶ Работу подключённого обработчика, уникальность ID, передачу координат.

Результаты:

- ✓ Все тесты пройдены на эмуляторе и реальном устройстве Аврора 2.0.
- ✓ Компонент полностью соответствует техническому заданию.

ПЕРЕХОД К ЗАКЛЮЧЕНИЮ И ИТОГАМ

Что сделано:

- ▶ Разработан готовый к использованию QML-компонент.
- ▶ Реализованы все функциональные требования.
- ▶ Обеспечено расположение поверх UI без сдвига.
- ▶ Подключаемый обработчик с координатами – уникальная особенность
- ▶ Подготовлен отчёт по ГОСТ, код опубликован в открытом репозитории.

ВЫВОДЫ

Основные выводы:

Системная клавиатура Авроры не подходит для задач, требующих контроля над вводом.

QML + Sailfish Silica – оптимальный выбор для быстрой разработки нативного компонента.

Использование Repeater и массивов раскладок позволяет легко масштабировать компонент.

Механизм активного поля и вставка с учётом курсора обеспечивают удобство ввода.

Подключаемый обработчик с уникальными ID и координатами даёт разработчику полный контроль над каждым нажатием.

Компонент переиспользуем и готов к внедрению в любые приложения для Авроры.