

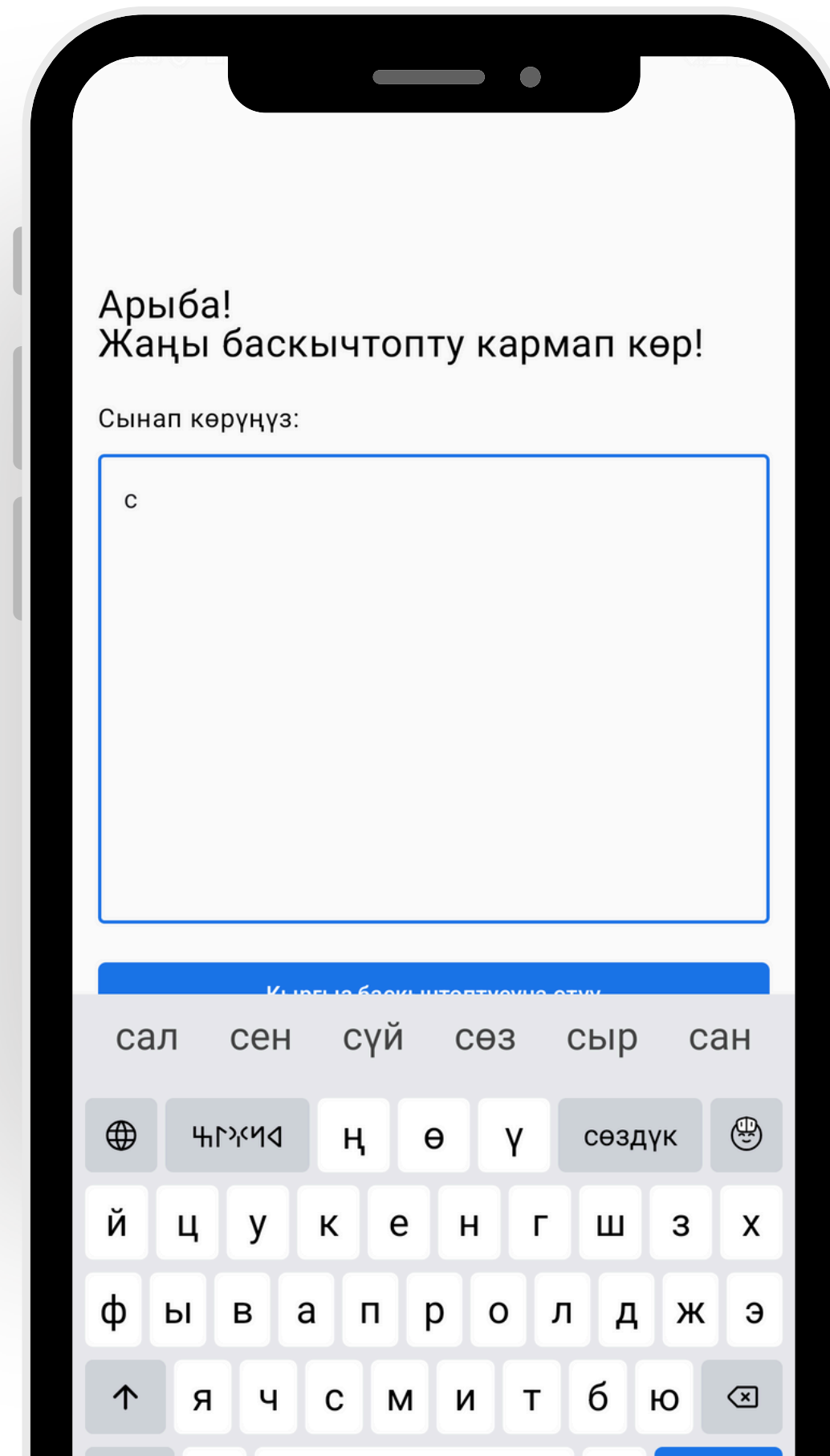
Kyrgyz Keyboard

Team Members:

Ademi Abdykerimova
Nikita Paniukhin
Pavel Boubel

Mentor:

Ivan Komarov



Domain Introduction

What is the domain?

Minority-language tools for underserved communities

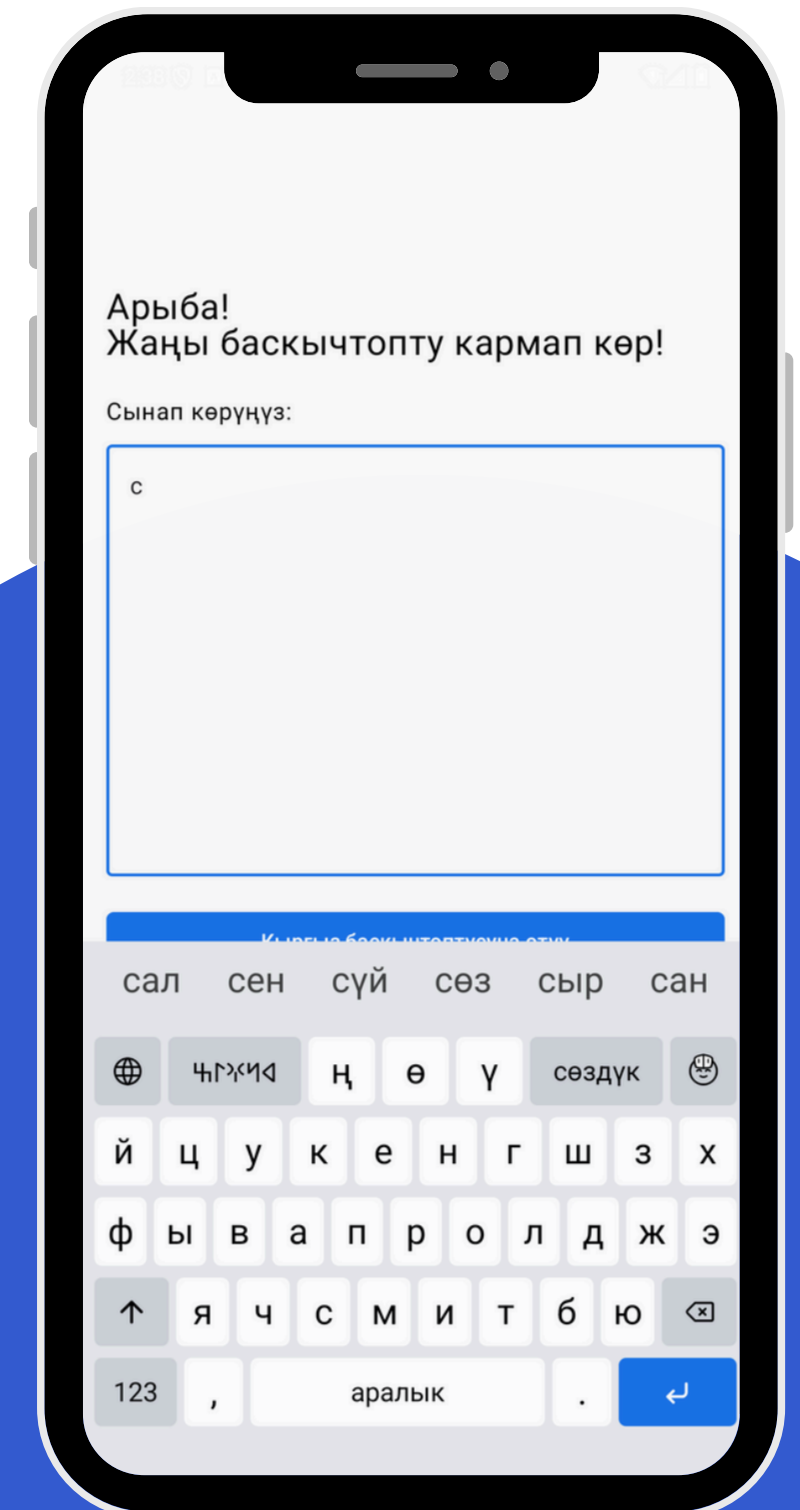
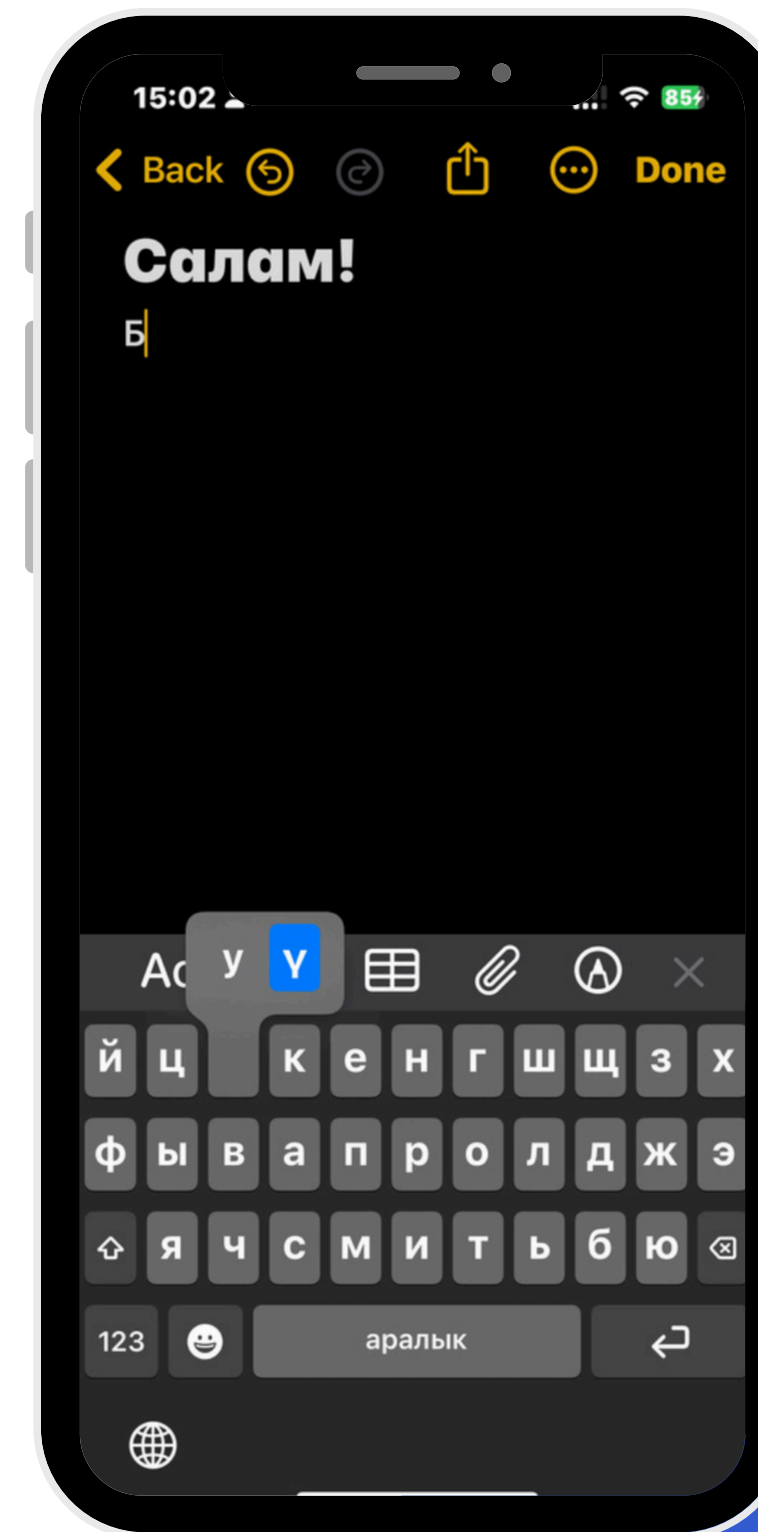
Why is this important?

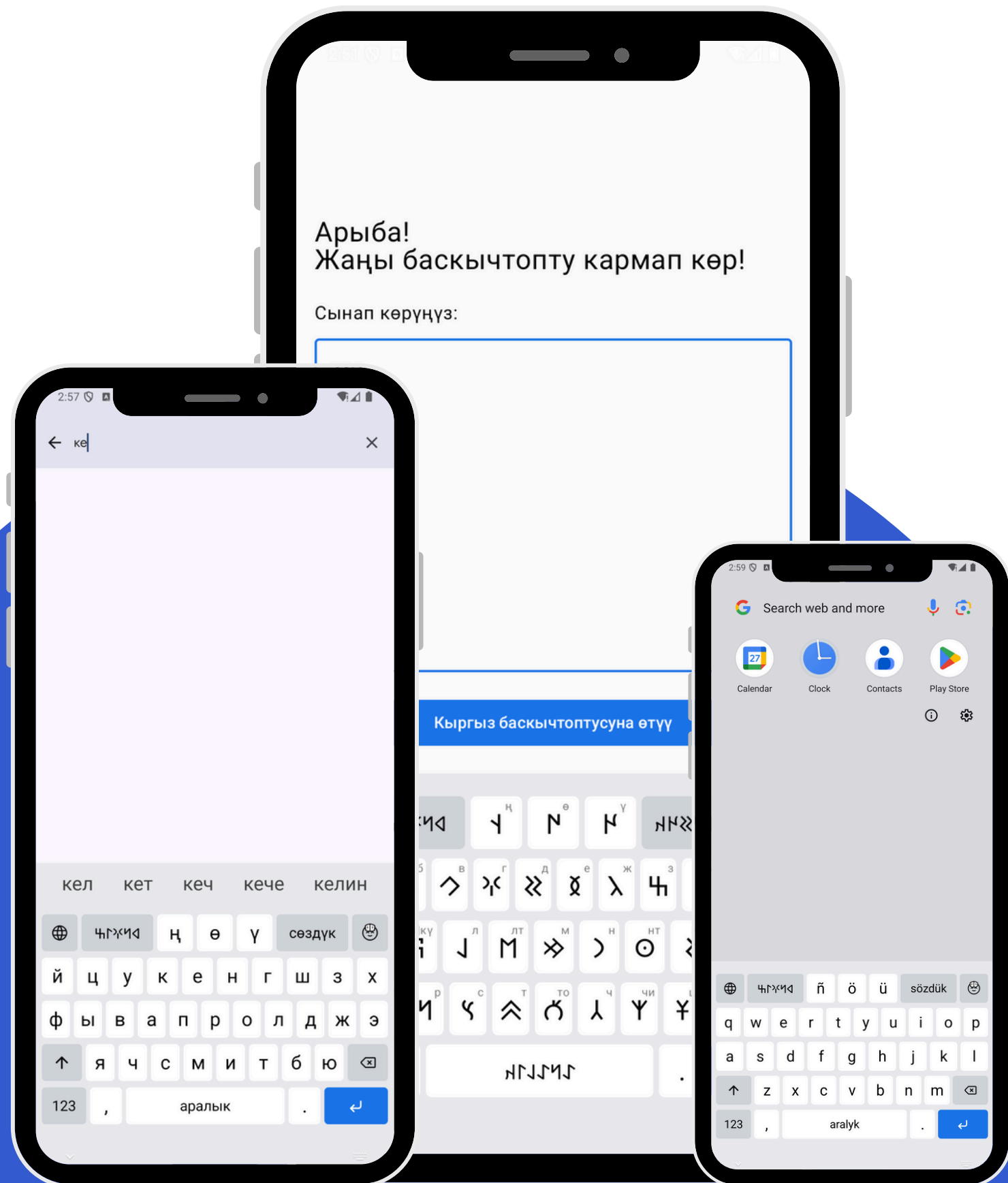
Existing Kyrgyz keyboards lack optimized layouts, predictive text, or script-switching (Cyrillic/Latin)

Who benefits?

Empowers students, elders, professionals to type naturally.
Preserves the Kyrgyz language in the digital space

First open-source keyboard tailored for Kyrgyz





Welcome To Our Application

— Bridging digital gaps for Kyrgyz speakers

Optimized Layout

All Kyrgyz letters (ң, ө, ү) accessible without long-press — type faster

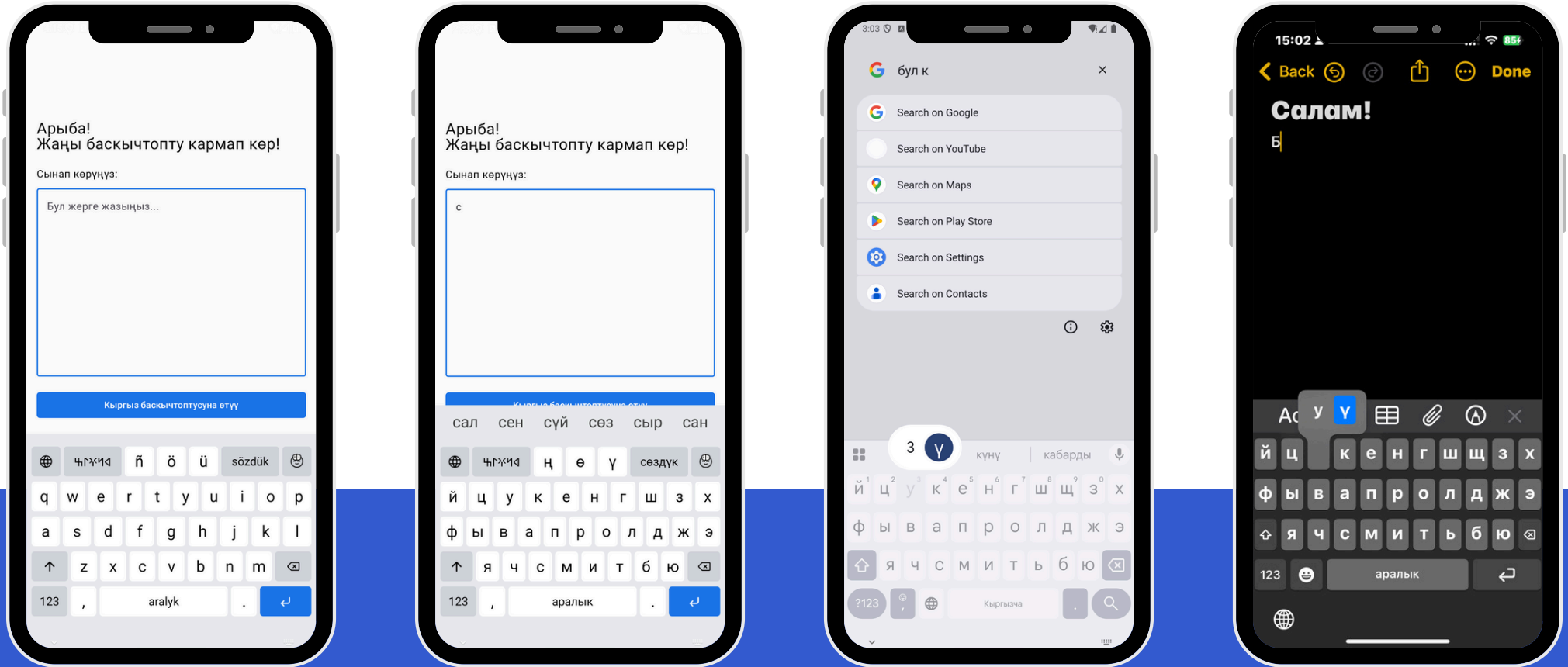
Word Predictions & Autocorrect

Reduces typos (e.g., fixes "курки" → "курут")
Real-time fixes for agglutinative word forms

User-Centric Design

Extra layouts and symbols to encourage engagement

Analogs & Comparison



Features	Our Keyboard	Gboard (Google)	iOS Keyboard	Samsung Keyboard
Kyrgyz Layout	✓ Ң ө ү	Long-press	Long-press	Long-press
Predictive Text for Kyrgyz	✓ Statistical model	Limited (basic)	✗	✗
Dialect Support	✓	✗	✗	✗

Technologies used



Kotlin for Android development

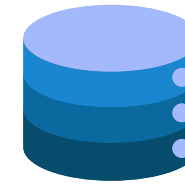
Jetpack Compose for modern UI

Material Design 3

ViewModel

StateFlow for state management

Coroutines for async operations



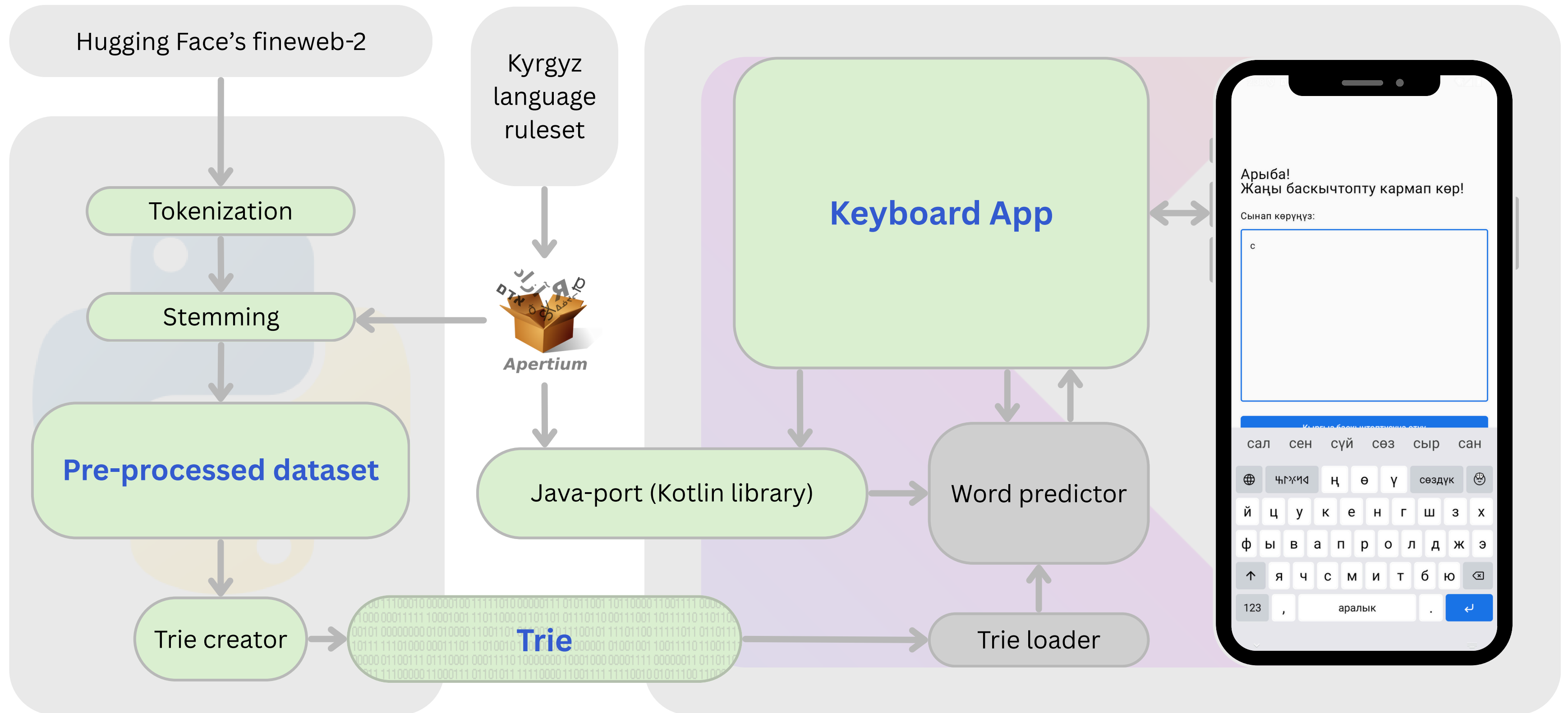
Python for highly parallel data processing

Hugging Face's **fineweb-2** dataset
for Kyrgyz language corpus

Apertium and **Helsinki Finite State Toolkit**
for morphological analysis and stemming

Trie structures for faster processing

Architecture



Ademi

Android App Development

Current Status:

Main app UI & Keyboard System complete

Core Implementation

- Custom Input Method Service
- Material Design 3 UI with Jetpack Compose
- Keyboard activation & permission flow
- System integration & settings management

Short-term goals

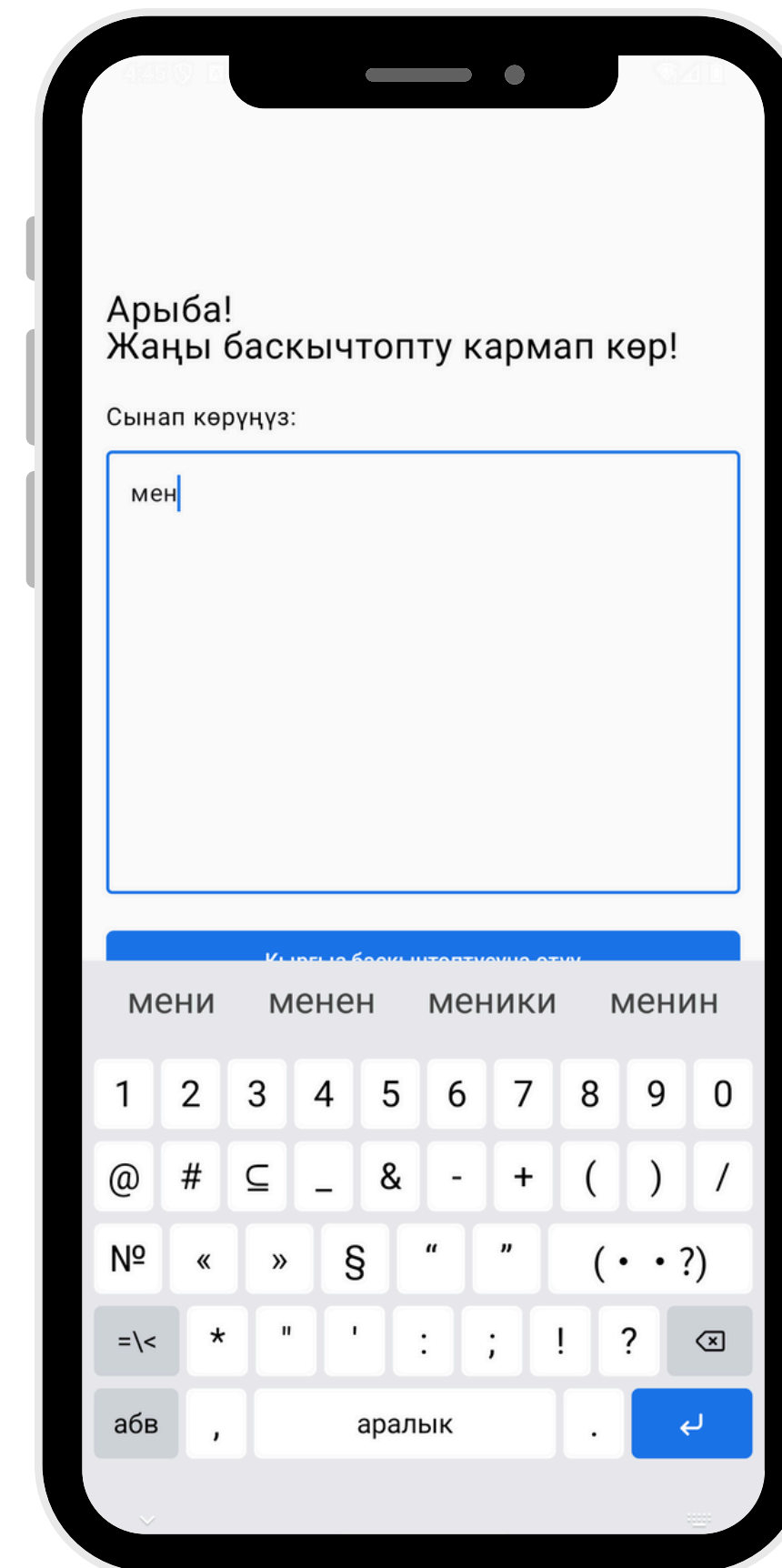
- Layout optimization
- APK size reduction
- Performance improvements
- Integration preparation for NLP team's predictive text

Technical Challenges Solved

- Complex keyboard activation workflow
- Created guided permission flow
- Implemented automatic status verification
- Built seamless keyboard switching

If Time Permits

- Theme customization
- Keyboard height adjustment
- Key vibration patterns



Nikita

— Datasets, Stemming, and Pre-processing

Current Status:

Prebuilt words and stems for creating and using prediction structures

Pre-processing

- Found the optimal source of data for the model
- Implemented a custom regex-based tokenizer for segmenting the corpus into words and sentences

Optimizations

- Wrote a complex workflow for highly parallel tasks, including multi-process computations and multi-threaded concurrent IO operations

Misc

- Researched methods of dictionary and trie compression

Word stemming

- Integrated and optimized 2 dictionaries for suffix extraction (currently deprecated)
- Integrated Apertium for offline (algorithm-wise) stemming

Stemming on mobile

- Compiled and linked an old version of Apertium's internals for usage on Android

Future work

- Research ways to base predictions on word stems and suffixes

Pavel

—— Word prediction (Trie) and Storage/Transfer on mobile

Current Status:

Ready-to-use predictor of words based on context

Solved tasks

- Prediction of text based on it's context (three most likeliest words)
- Uploading new texts (so that we can keep the predictor up-to-date with current grammar)

CLI app

- Smoothy way to test updates of predictor
- Support two types of operations: loading the new text and getting hints of next words to the current context
- Fully implemented in Python

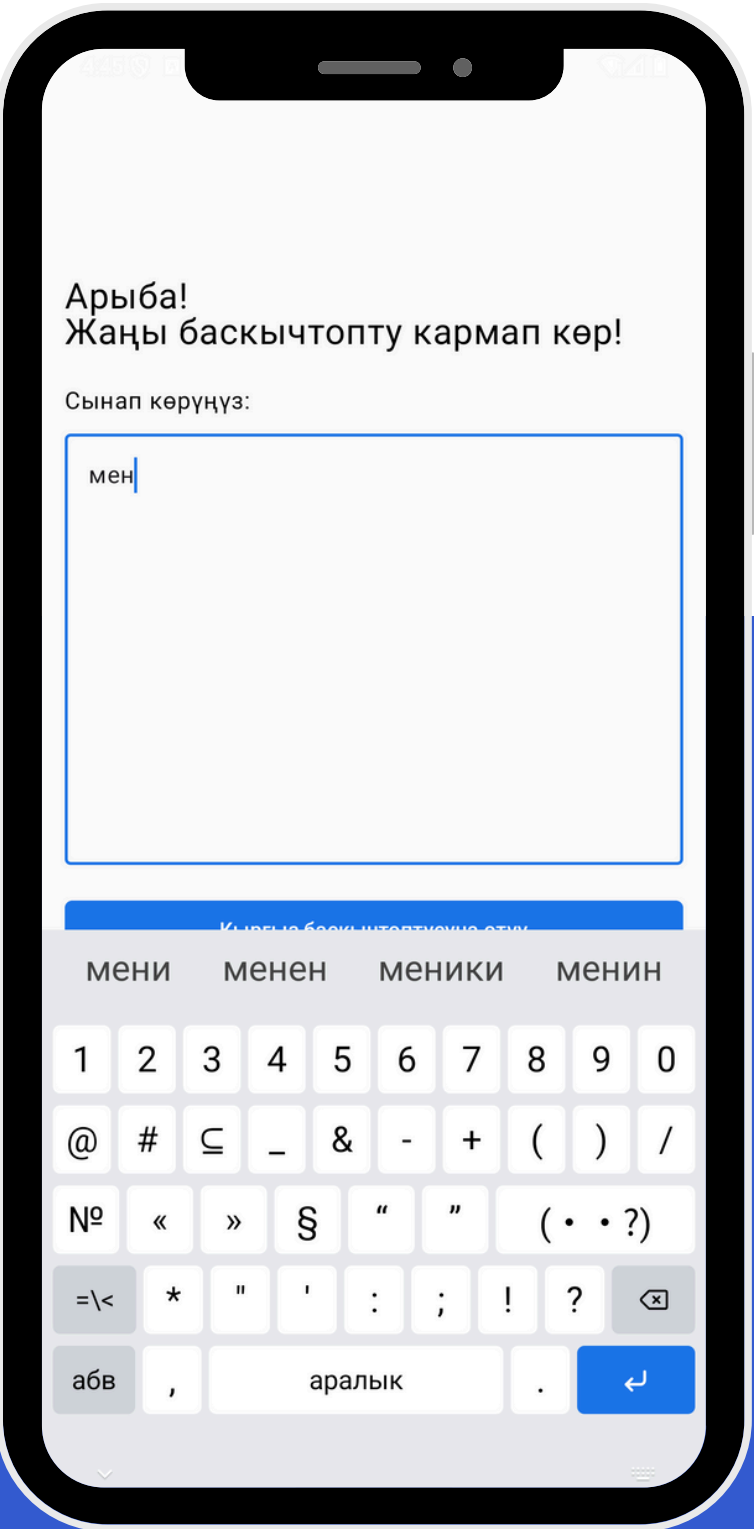
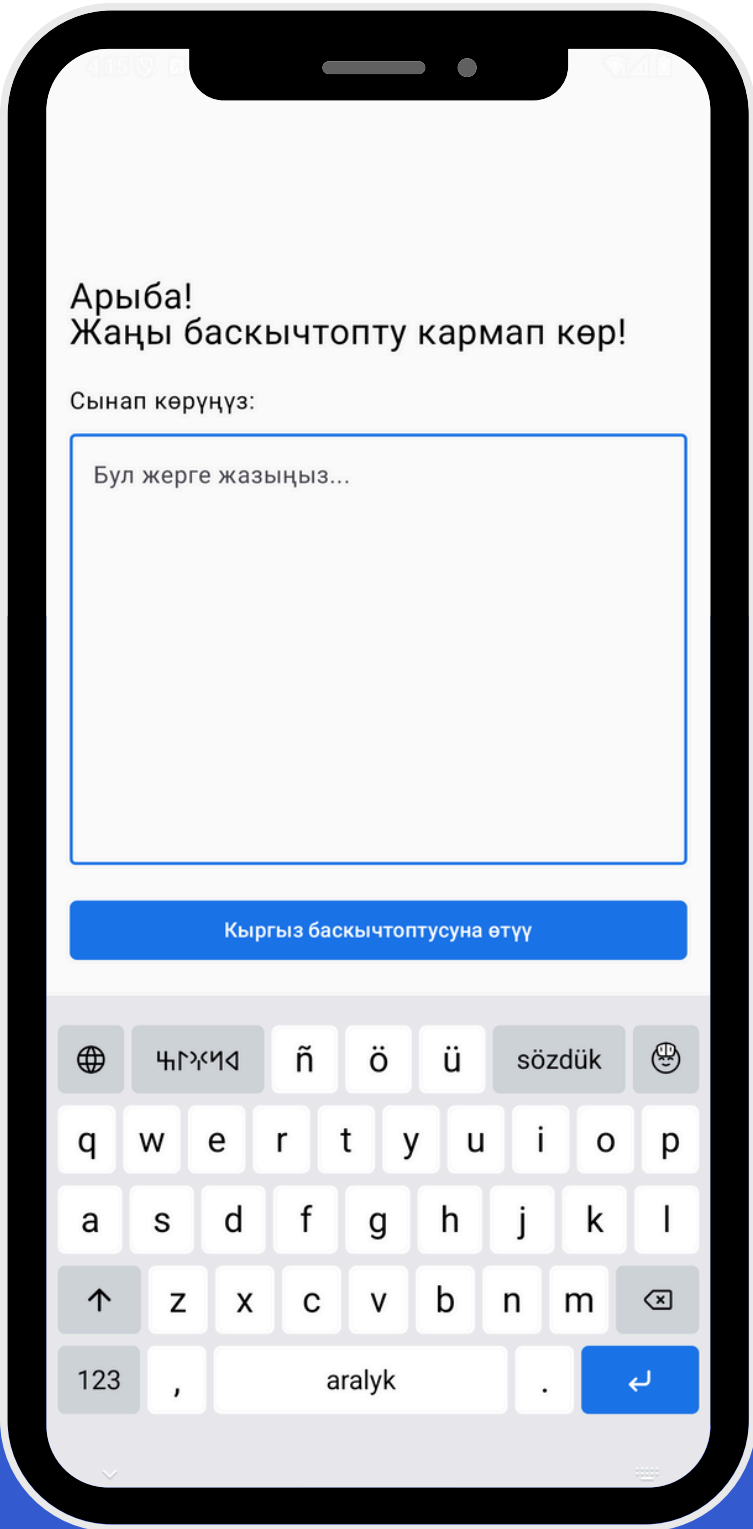
Current challenges

- Optimising memory (possible way to solve — hashing of words instead of keeping them all the way in the trie)
- Working on better predictions of next word based on current context

Tasks for future

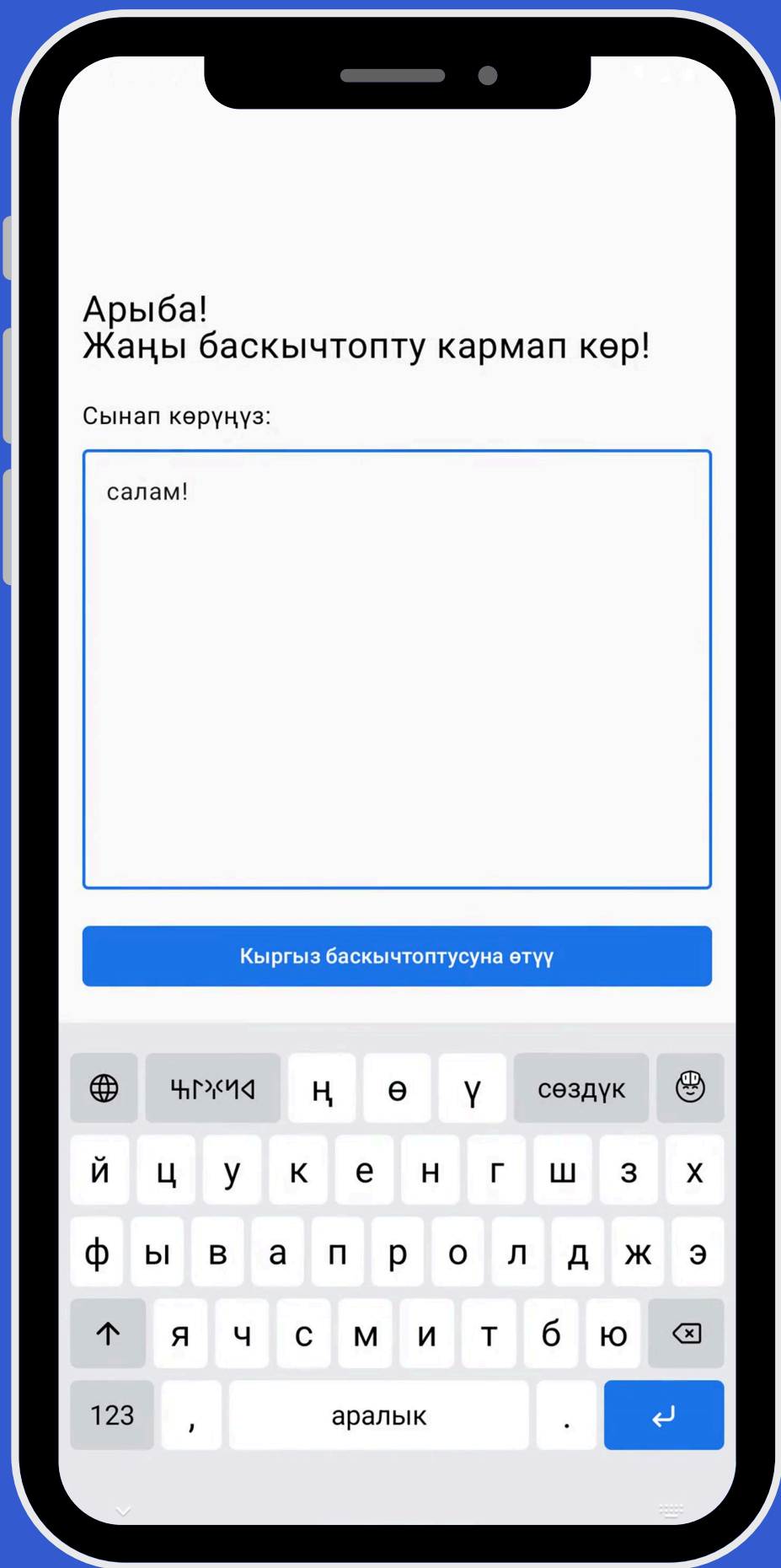
- Support prediction of the ending of the word based on it's beginning
- Convenient transfer to mobile

Screenshots



Prediction examples

Prompt	Completion		
and ...	the	a	his
in different ...	parts	places	countries
years old ...	when	and	at
length of ...	river	body	superpermutation
may choose to ...	use	attend	still



Demo

<https://github.com/Kyrgyz-Keyboard>

<https://kyr.npanuhin.me/Kyrgyz-Keyboard.apk>