

[FAQ](#)[Getting Started](#) >[Quick Start](#)[Introduction](#) v[Advanced Installation](#)[Integration with React](#)[TL](#) >

## Integration with React

In this article, we will look at the combination of gramjs and React

After creating the project and installing all the necessary dependencies, you can use gramjs in your projects

### Table of Contents

- [Client launch](#)
- [Save session](#)
- [Catching errors](#)

### Client launch

In the example below you will see an example of an application that sends a confirmation code and starts the client

[JavaScript](#) [TypeScript](#)

```
import React, { useState } from 'react'
import { TelegramClient } from 'telegram'
import { StringSession } from 'telegram/sessions'

const SESSION = new StringSession('')
const API_ID = 00000000 // put your API ID here
const API_HASH = '111eb4dc492' // put your API HASH here

const client = new TelegramClient(SESSION, API_ID, API_HASH)

const initialState = { phoneNumber: '', password: '' }

export function App () {
  const [{ phoneNumber, password }, setAuthInfo] = useState(initialState)

  async function sendCodeHandler () {
    await client.connect() // connect to the server
    await client.sendCode({
      apiId: API_ID,
      apiHash: API_HASH,
      phoneNumber
    })
  }

  async function clientStartHandler () {
    await client.start({ phoneNumber, password })
    await client.sendMessage('')
  }

  function inputChangeHandler (e) {
    setAuthInfo({authInfo: e.target.value}) =>
  }

  function userAuthParamCallback () {
    return async function () {
      return await new Promise((resolve, reject) => {
        resolve(param)
      })()
    }
  }

  return (
    <div>
      <input
        type="text"
        name="phoneNumber"
        value={phoneNumber}
        onChange={inputChangeHandler}
      />

      <input
        type="text"
        name="password"
        value={password}
        onChange={inputChangeHandler}
      />

      <input type="button" value="Send Code" />

      <input
        type="text"
        name="phoneCode"
        value={phoneCode}
        onChange={inputChangeHandler}
      />
    </div>
  )
}
```