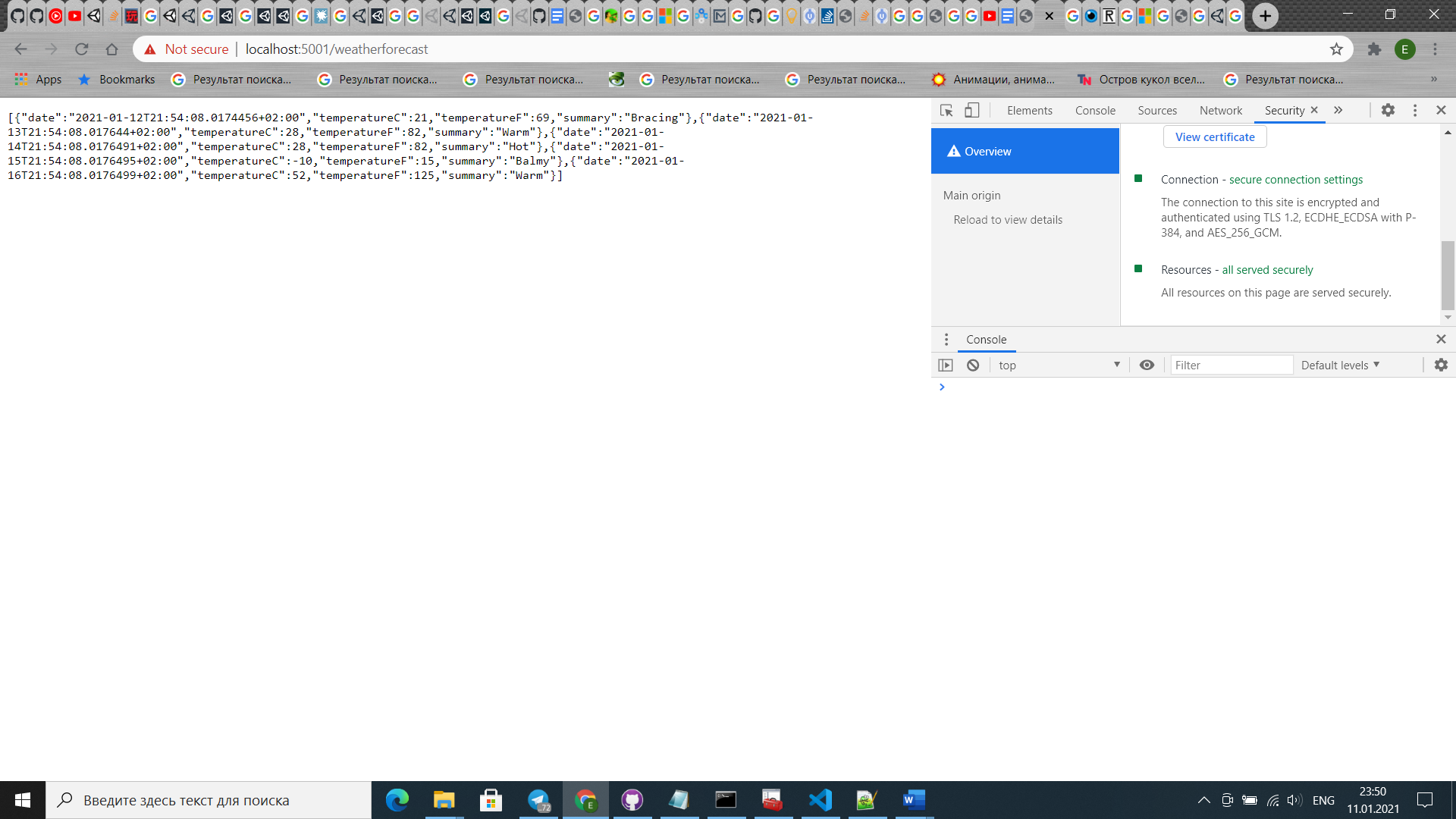
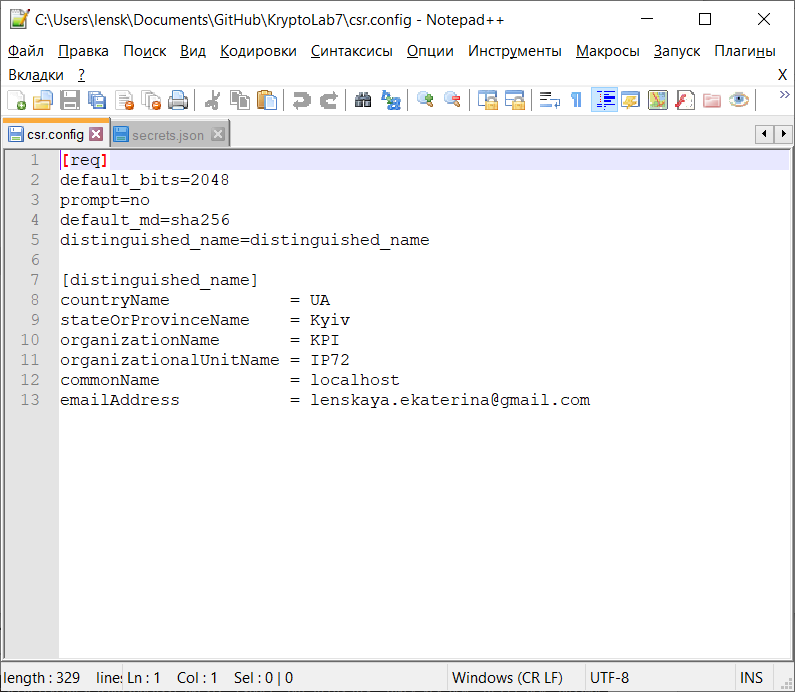
# Report

TLS 1.2, ECDHE\_ECDSA with P-384 and AES\_256\_GCM were used as a recommended cipher suite.

ECDHE (Elliptic Curve Diffie-Hellman Ephemeral) and ECDSA (Elliptic Curve Digital Signature Algorithm) were chosen because elliptic curves remain mathematically strong. ECDHE allows two parties, each having an elliptic curve public-private key pair, to establish a shared secret over an insecure channel. AES\_256\_GCM is used to exchange messages later.



Configuration of a certificate:



Generation of the certificate and PFX-file:

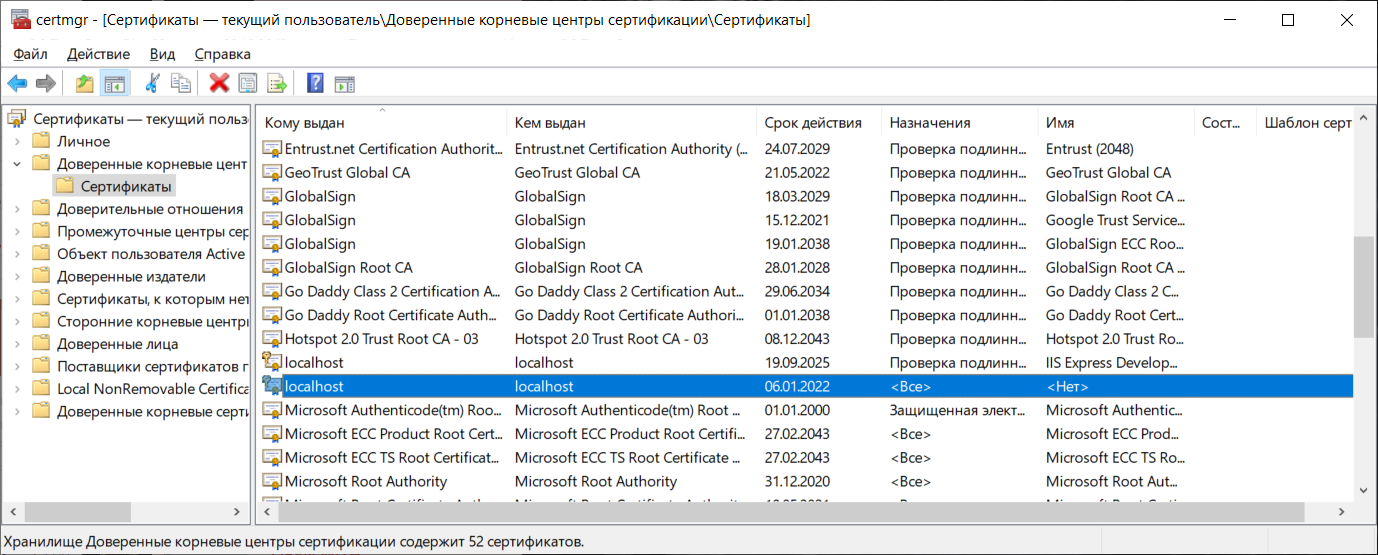
openssl ecparam -genkey -name prime256v1 -out key.pem

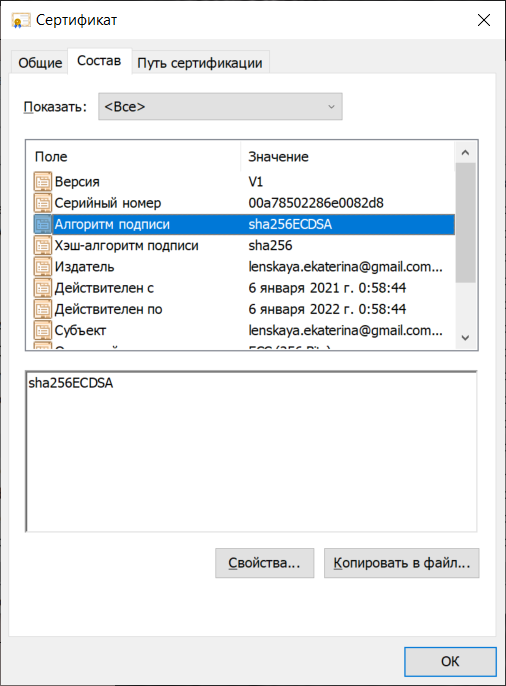
openssl req -new -sha256 -key key.pem -out csr.pem -config csr.config

openssl x509 -req -in csr.pem -signkey key.pem -out cert.pem -days 365 -sha256

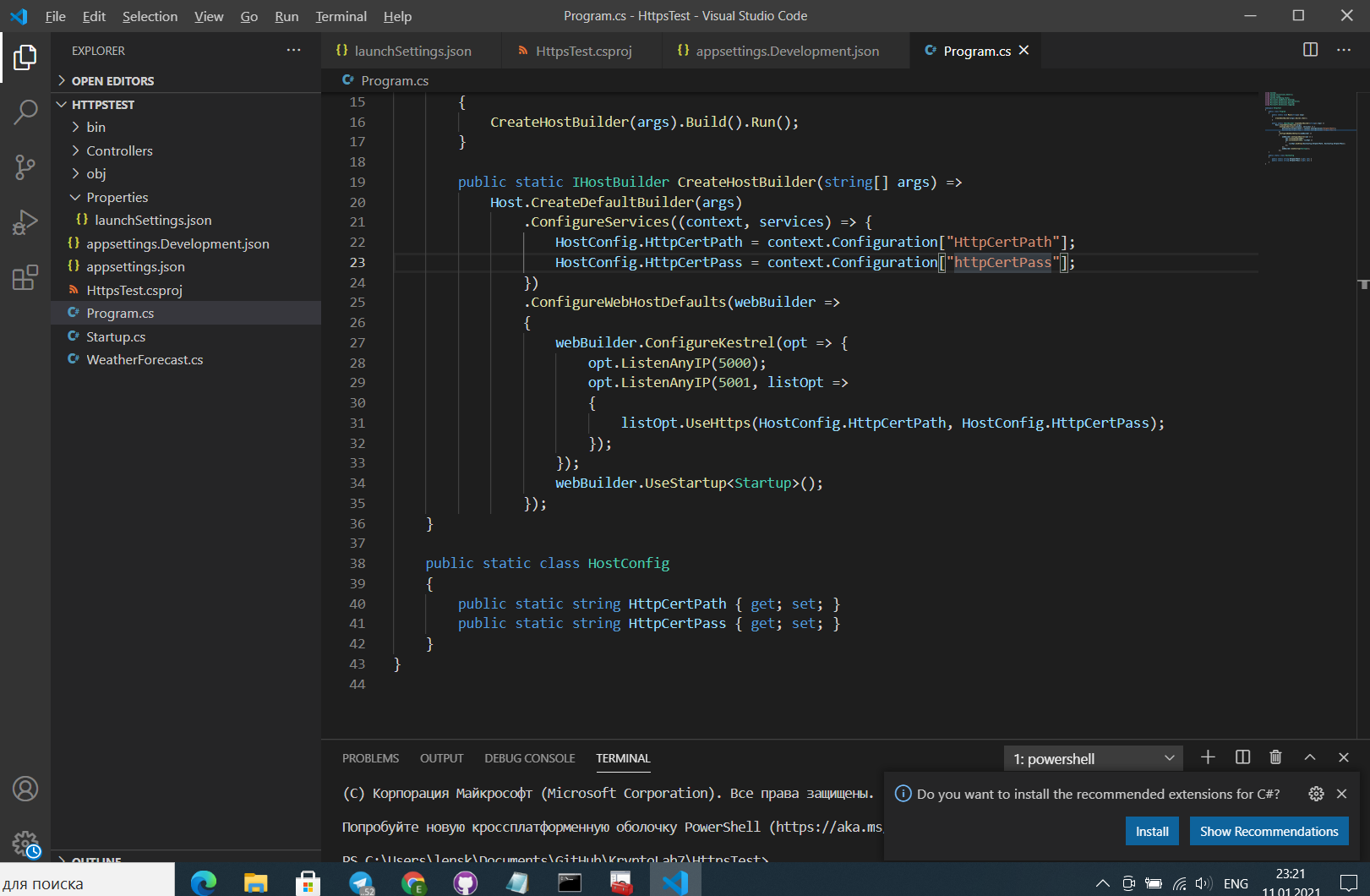
openssl pkcs12 -export -out httpCert.pfx -inkey key.pem -in cert.pem -password pass: 7N9qsMgcj7j88y57

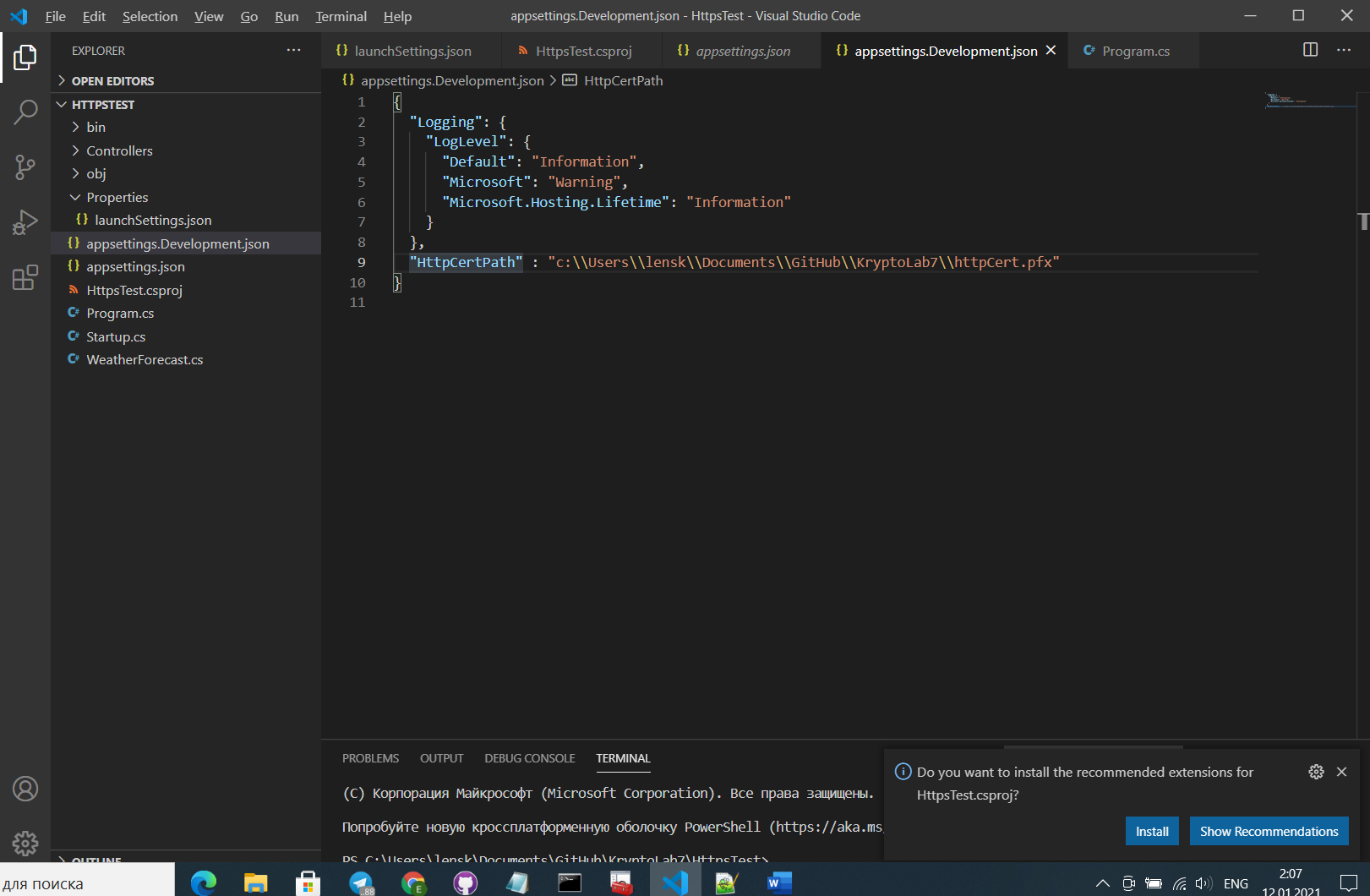
Certificate was added to trusted list:

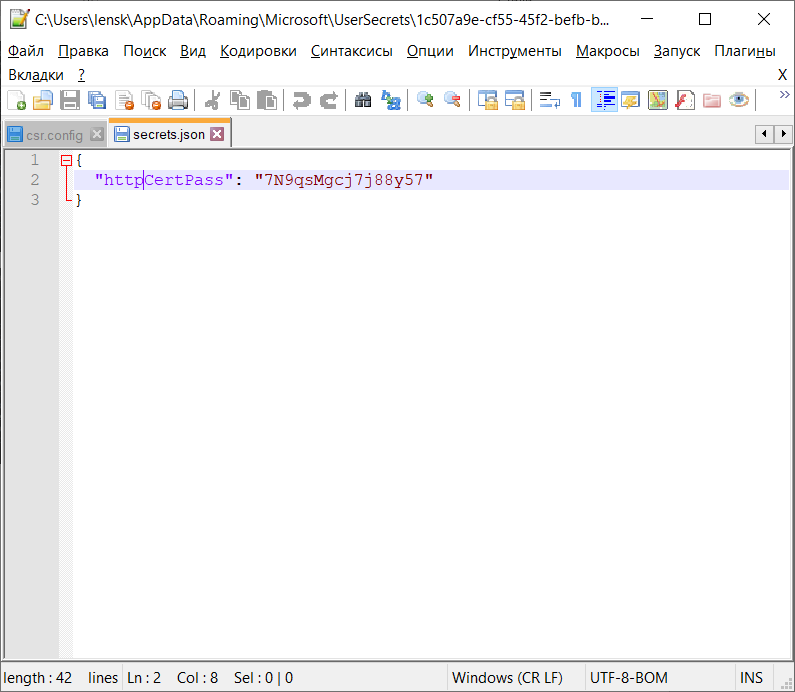




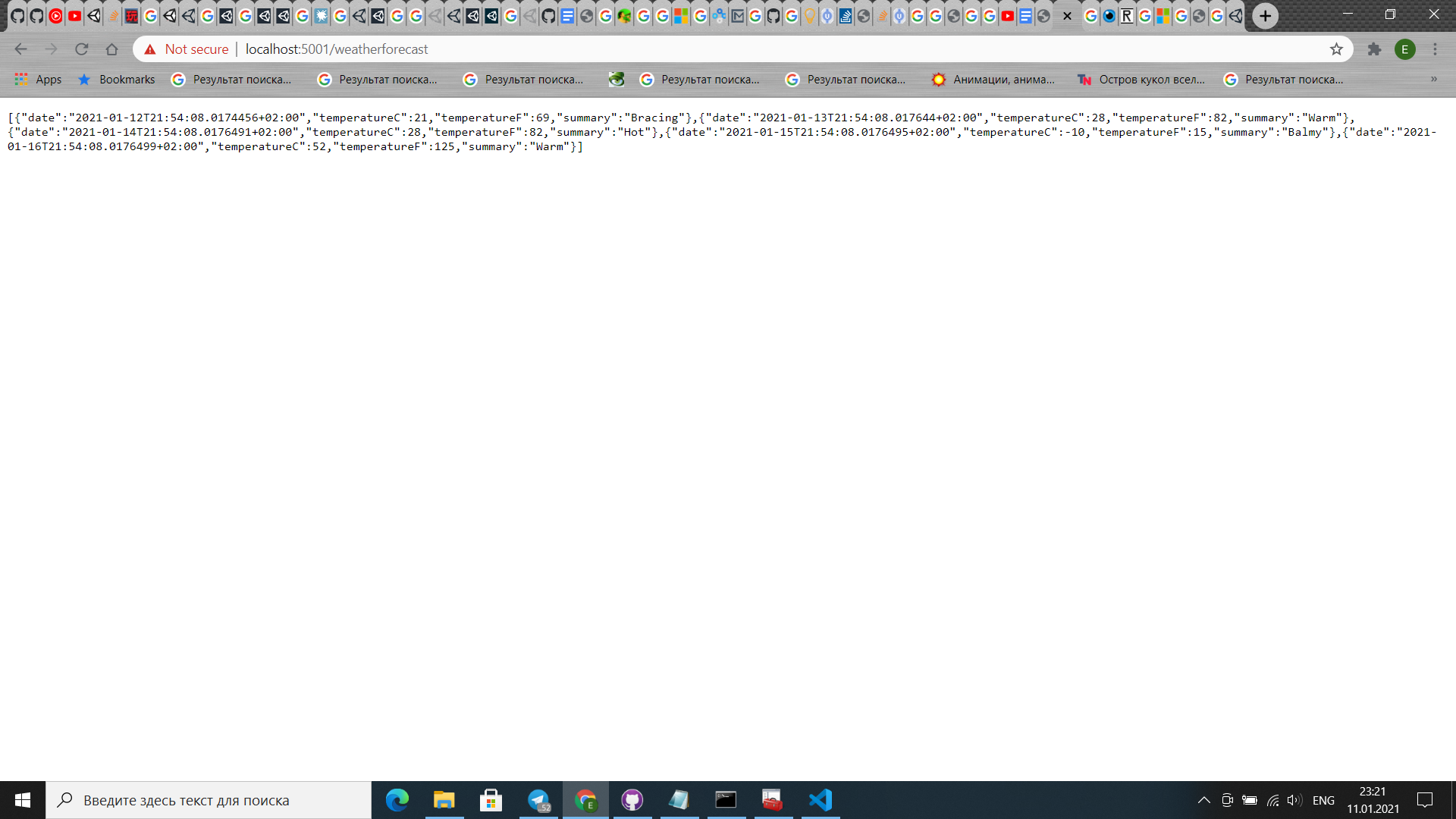
Configuration of certificate in code:







Result:



Browser still identifies connection as not secure because certificate was given to us by ourselves. Intermediate certificate is needed to make browser trust this connection.