ChainSafe Gaming SDK

WebGL

Get User's Network

```
/*
1 Mainnet
3 Ropsten
4 Rinkeby
5 Goerli
42 Kovan
56 Binance Smart Chain Mainnet
97 Binance Smart Chain Testnet
100 xDai
137 Matic
1287 Moonbase Testnet
80001 Matic Testnet
43113 Avalanche Testnet
43114 Avalanche Mainnet
*/
int networkId = Web3GL.Network();
```

Send Transaction through WebGL

```
// account to send to
string to = "0x428066dd8A212104Bc9240dCe3cdeA3D3A0f7979";
// amount in wei to send
string value = "12300000000000000";
// gas limit OPTIONAL
string gasLimit = "";
// gas price OPTIONAL
string gasPrice = "";
// connects to user's browser wallet (metamask) to send a transaction
try {
    string response = await Web3GL.SendTransaction(to, value, gasLimit, gasPrice);
    Debug.Log(response);
} catch (Exception e) {
    Debug.LogException(e, this);
}
```

Send Contract through WebGL

Send will execute a smart contract method, altering the smart contract state.

Working example: https://chainsafe.github.io/game-sendContract-example/

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract AddTotal {
    uint256 public myTotal = 0;

    function addTotal(uint8 _myArg) public {
        myTotal = myTotal + _myArg;
    }
}
```

```
// smart contract method to call
string method = "addTotal";
// abi in json format
string abi = "[ { \"inputs\": [ { \"internalType\": \"uint8\", \"name\": \"_myArg\", \"typ
// address of contract
string contract = "0x7286Cf0F6E80014ea75Dbc25F545A3be90F4904F";
// array of arguments for contract
string args = "[\"1\"]";
// value in wei
string value = "0";
string gasLimit = "";
// gas price OPTIONAL
string gasPrice = "";
// connects to user's browser wallet (metamask) to update contract state
try {
  string response = await Web3GL.SendContract(method, abi, contract, args, value, gasLimit
  Debug.Log(response);
} catch (Exception e) {
  Debug.LogException(e, this);
}
```

Sign through WebGL

Working example: https://chainsafe.github.io/game-sign-example/

```
try {
  string message = "hello";
  string response = await Web3GL.Sign(message);
  Debug.Log(response);
} catch (Exception e) {
  Debug.LogException(e, this);
}
```