

Kvantni novac

Vanja Kovinić 42/20 RN





Kvantni novac

Cilj projekta:

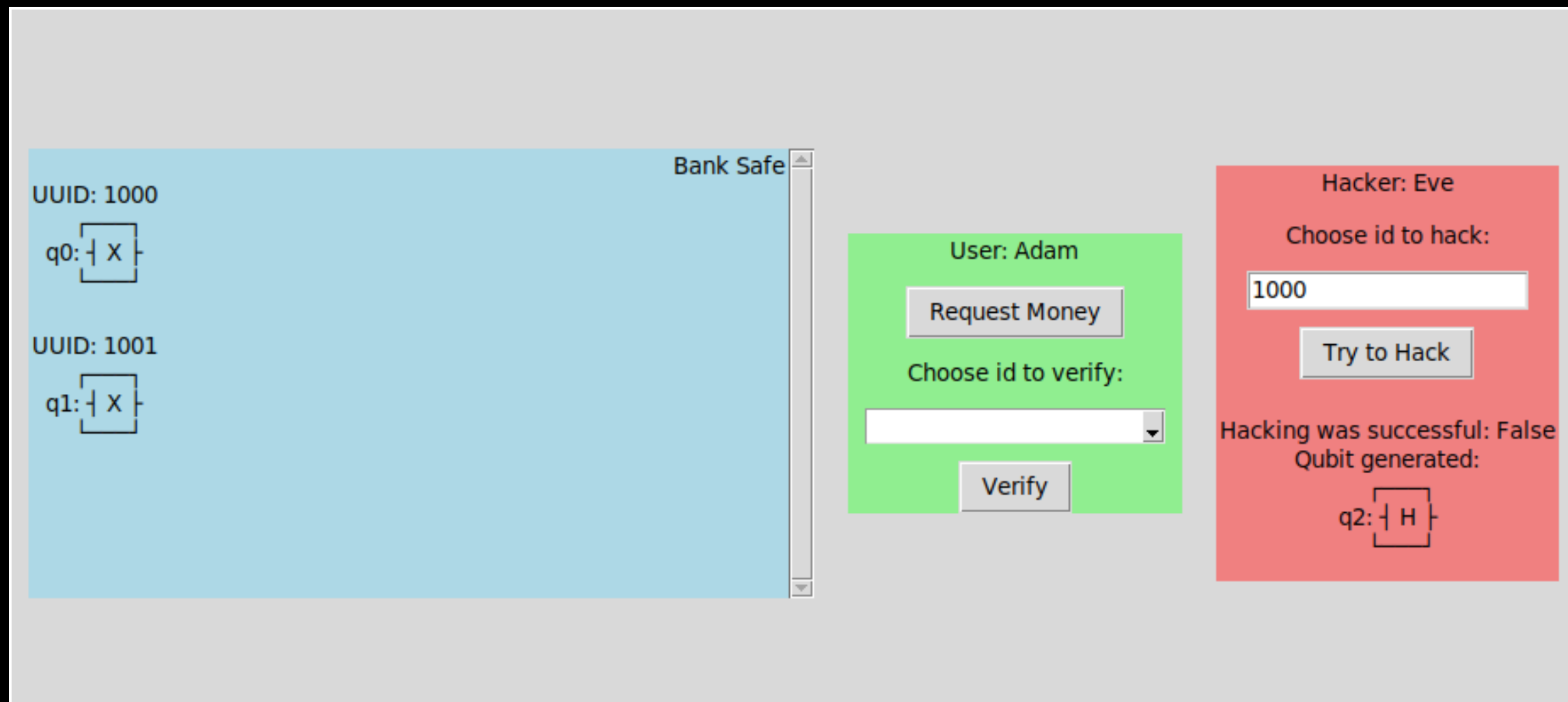
- Unapređenje sigurnosti novčanica korišćenjem kvantnih koncepta i tehnologija.

Ključne tačke:

Generisanje nasumičnog stanja kubita korišćenjem kvantnog algoritma za generisanje nasumičnih brojeva

Postavljanje kubita u jedno od predefinisanih stanja korišćenjem nasumičnog broja





Korisnički interfejs

Klasni dijagram

Quantum_Bank

Polja:

- storage_path
- n_quantum_states
- current_uuid
- storage

Metode:

- create_money
- issue_money
- verify_money
- store_money
- laod_from_storage
- _true_random

User

Polja:

- name
- bank
- money

Metode:

- request_money
- verify_money

Hacker

Polja:

- name

Metode:

- hack
- qc_generator

Delovi koda

Funkcija za generisanje
slučajnog stanja vektora

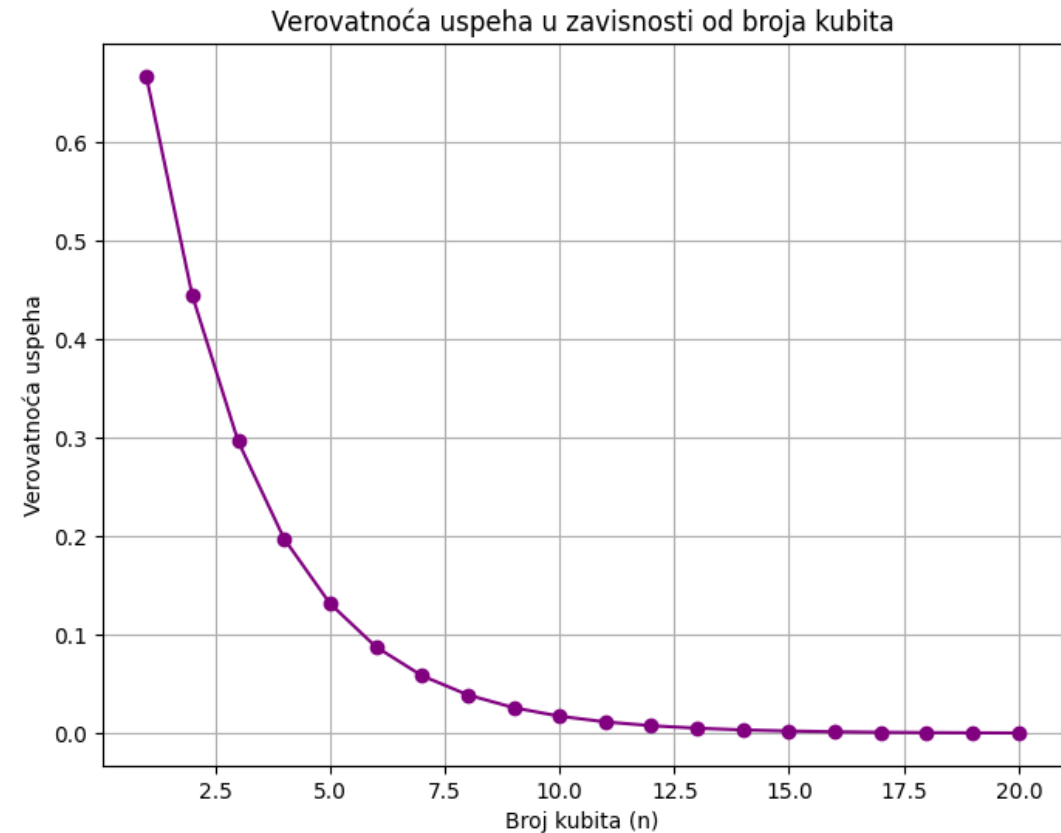
```
def _true_random(self, n_numbers=1):  
  
    # Calculate the number of qubits needed  
    n_qubits = (self.n_quantum_states - 1).bit_length()  
  
    # Create a quantum register of n qubits  
    qreg = QuantumRegister(n_qubits)  
    qc = QuantumCircuit(qreg)  
  
    # Apply Hadamard gates to all qubits  
    qc.h(qreg)  
  
    # Measure all qubits  
    qc.measure_all()  
  
    # Simulate the circuit  
    simulator = Aer.get_backend('qasm_simulator')  
    job = assemble(transpile(qc, simulator), shots=n_numbers)  
    result = simulator.run(job).result()  
    counts = result.get_counts()
```

Funkcija za generisanje nasumičnih brojeva

```
true_randoms = self._true_random(n_qubits)  
  
for i in range(n_qubits):  
    # |0>  
    if true_randoms[i] == 0:  
        pass  
  
    # |1>  
    elif true_randoms[i] == 1:  
        qc.x(qreg[i])  
  
    # |+>  
    elif true_randoms[i] == 2:  
        qc.h(qreg[i])  
  
    # |->  
    elif true_randoms[i] == 3:  
        qc.x(qreg[i])  
        qc.h(qreg[i])  
  
    # |i+>  
    elif true_randoms[i] == 4:  
        qc.h(qreg[i])  
        qc.s(qreg[i])  
  
    # |i->  
    elif true_randoms[i] == 5:  
        qc.x(qreg[i])  
        qc.h(qreg[i])  
        qc.s(qreg[i])
```

Testovi

- Što više budemo pokretali testove, verovatnoće da haker pogodi sva stanja kubita, težiće funkciji na datom prikazu





Zaključak

Neke od stavki za sledeću iteraciju su:

- Rad sa više kubita
- Uvođenje novih stanja u kojima se kubit može naći
- Korišćenje protokola za enkripciju serijskog broja
- Korišćenje naprednijih tipova skladištenja podataka

