Implementacija funkcionalnosti generatora paketa navedenog u uputama napisana je u tri skripte:

* Receiver.py
* Generator.py
* MarkovChain.py

Zbog ograničenja mreže u kojoj smo obavljali implementaciju (Eduroam), nismo bili u mogućnosti slati pakete preko lokalne mreže, već smo pakete slali na loopback adresu (127.0.0.1).

Skripta Receiver.py implementira jednostavnog primatelja UDP paketa. Sluša na vratima 8080, prima UDP pakete te ispisuje njihovu veličinu na konzolu. Izvorni kod je na slici 1. Također prikazana je obrada najčešćih pogrešaka (opis u komentarima koda).

A screen shot of a computer program

Description automatically generated

Slika 1. Receiver.py skripta

Skripta Generator.py implementira generator prometa. U kratko ova skripta generira pakete različitih veličina i među dolaznih vremena, a sve na osnovu nasumičnih vrijednosti (dobivenih iz odgovarajućih, analizom dobivenih, distribucija).

A screen shot of a computer

Description automatically generated

Slika 2. Generator.py Formiranje i slanje paketa određene veličine

A screen shot of a computer program

Description automatically generated

Slika 3. Generator.py Generiranje paketa za odabranu vrstu usluge

Kao što je prikazano na slici 3, ovisno o vrsti usluge generira se slučajna vrijednost veličine paketa te među dolazno vrijeme (delay) na osnovu odgovarajućih kontinuiranih distribucija (dobivene analizom).  
Nakon što se odrede vrijednosti veličine paketa i vremena uspavljuje se dretva na broj milisekundi definirano u varijabli delay, te se poziva funkcija za formiranje i slanje paketa s dobivenom vrijednosti.

Unutar skripte MarkovChain.py implementiran je automat stanja (Markovljev lanac). Sastoji se od 3 stanja: Agario, Radio i Video. Na slici 4. su prikazane i dodatne pomoćne funkcije.

A screenshot of a computer program

Description automatically generated

Slika 4. MarkovChain.py Definicija stanja, vjerojatnosti prijelaza te pomoćne funkcije

Na slici 5. prikazana je implementacija automata. Počinje u nasumičnom stanju. Zatim za trenutno stanje se izvode sljedeće akcije:

* Računa se vrijeme ostanka u stanju na osnovu eksponencijalne razdiobe
* Za dobiveni broj sekundi se izvodi funkcija do\_action\_for\_seconds koja za određeno stanje izvodi odgovarajuće generiranje paketa
* Na osnovu matrice prijelaza se računa sljedeće stanje

Akcije se izvode u beskonačnoj petlji (do prekida programa).

A screen shot of a computer program

Description automatically generated

Slika 5. Implementacija stanja i akcija