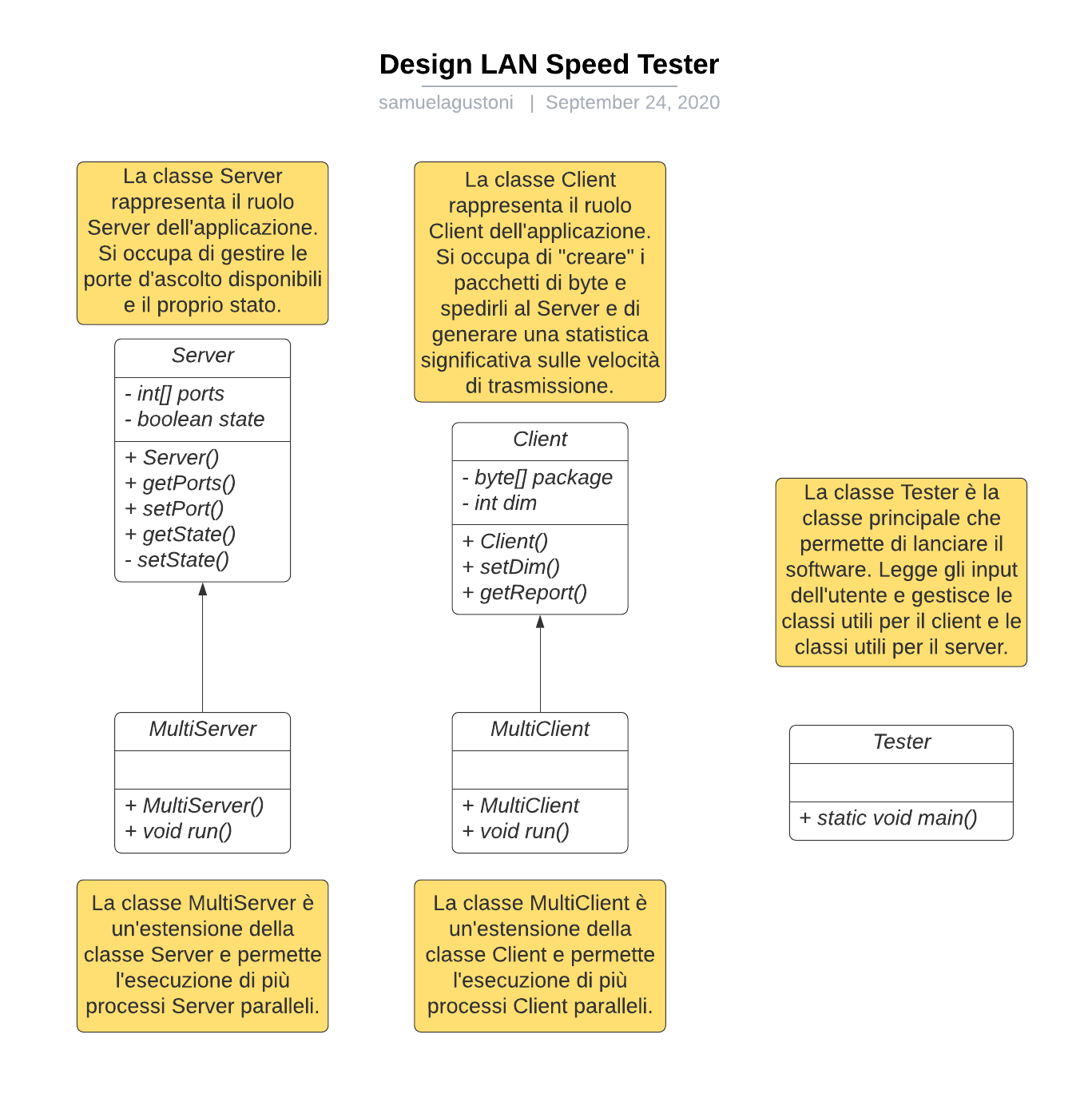
# Design della logica LAN Speed Tester

## Server class

### Attributes:

* **private int[] ports**: è l’array contenente le porte libere utilizzabili dal server per mettersi in ascolto e ricevere pacchetti di byte.
* **private boolean state**: è lo stato del server, se true significa che il server è pronto e in ascolto, se false non è pronto.

### Methods:

* **public Server()**: è il costruttore della classe, istanzia un oggetto Server.
* **public void getPorts()**: cerca le porte disponibili e le inserisce nell’attributo “ports”.
* **public void setPorts()**: si occupa di impostare la porta sulla quale il server rimarrà in ascolto.
* **public void getState()**: si occupa di riferire all’utente se il server è pronto e in ascolto.
* **private setState()**: imposta lo stato del server.

## Client classH:\LANSpeedTester\Doc\DesignLANSpeedTester.png

### Attributes:

* **private byte[] package**: l’array di byte da spedire al server, le sue dimensioni sono definite dall’utente.
* **private int dim**: è la dimensione dell’array di byte.

### Methods:

* **public Client()**: è il costruttore della classe, istanzia un oggetto Client.
* **public void setDim()**: permette all’utente di impostare la dimensione dell’array package.
* **public void getReport()**: mostra all’utente una statistica significativa delle velocità di comunicazione tra client e server.

## MultiServer class

### Attributes:

### H:\LANSpeedTester\Doc\DesignLANSpeedTester.pngMethods:

* **public MultiServer()**: è il costruttore della classe, istanzia un oggetto MultiServer.
* **public void run()**: è utile per eseguire più thread alla volta.

## H:\LANSpeedTester\Doc\DesignLANSpeedTester.pngMultiClient class

### Attributes:

### Methods:

* **public MultiClient()**: è il costruttore della classe, istanzia un oggetto MultiClient.
* **public void run()**: è utile per eseguire più thread alla volta.

## H:\LANSpeedTester\Doc\DesignLANSpeedTester.pngTester class

### Attributes:

### Methods:

* **public static void main()**: si occupa di gestire tutte le interazioni con l’utente e mette in relazione le classi client e server.