



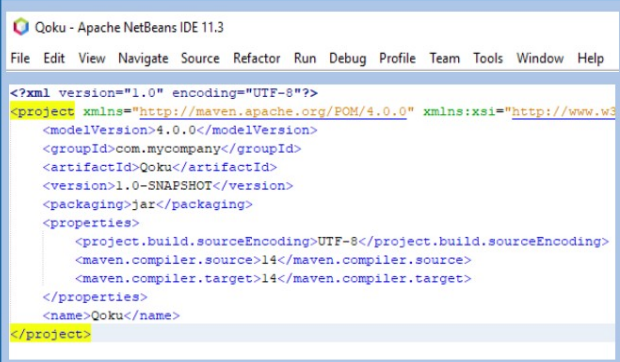

## ISTITUTO TECNICO INDUSTRIALE STATALE ANTONIO MEUCCI

Via del Filarete, 17 - 50143 - Firenze - Tel. 055 70.70.11 - Fax 055 71.08.76 - Email: [FITF010003@istruzione.it](mailto:FITF010003@istruzione.it) - PEC: [FITF010003@pec.istruzione.it](mailto:FITF010003@pec.istruzione.it)  
Codice Meccanografico: FITF010003 - Codice Fiscale: 80020810489 - Codice IBAN: IT92 0300 6902 9141 0000 0046 018 - CC postale: 27947506  
Codice TU: 0311957 - Codice IPA: istsc\_fitf010003 - Codice Univoco Ufficio FE: UF3850

### Pianificazione del progetto Chat multiutente one to one

**Dati Project Manager**  
Qafaloku Leone 5 B IA

**Data Inizio Progetto**  
02-11-2020

Versione Programma utilizzato	Elementi richiesti al fine di rispettare le richieste
 <p>Qoku - Apache NetBeans IDE 11.3</p> <pre>&lt;?xml version="1.0" encoding="UTF-8"?&gt; &lt;project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"&gt;   &lt;modelVersion&gt;4.0.0&lt;/modelVersion&gt;   &lt;groupId&gt;com.mycompany&lt;/groupId&gt;   &lt;artifactId&gt;Qoku&lt;/artifactId&gt;   &lt;version&gt;1.0-SNAPSHOT&lt;/version&gt;   &lt;packaging&gt;jar&lt;/packaging&gt;   &lt;properties&gt;     &lt;project.build.sourceEncoding&gt;UTF-8&lt;/project.build.sourceEncoding&gt;     &lt;maven.compiler.source&gt;14&lt;/maven.compiler.source&gt;     &lt;maven.compiler.target&gt;14&lt;/maven.compiler.target&gt;   &lt;/properties&gt;   &lt;name&gt;Qoku&lt;/name&gt; &lt;/project&gt;</pre>	 <p><b>Chat multiutente one to one</b></p> <p>Claudio Benvenuti • 26 ott (Ultima modifica: 27 ott)</p> <p>100 punti <span>Scadenza: 8 nov, 23:59</span></p> <p>Partendo dal progetto chat, documentare e realizzare i seguenti nuovi sviluppi:</p> <ol style="list-style-type: none"><li>1) gestione della disconnessione del client</li><li>2) i client devono conoscere i nomi dei client connessi</li><li>3) possibilità di avviare chat one to one oppure one to all</li></ol> <p>Commenti sul corso</p> <p> Aggiungi commento per il corso...</p>

#### Fase Del progetto:

1. Avendo già la versione precedente del progetto mi sono organizzato ad evolverla aggiungendo le seguenti linee di codice:

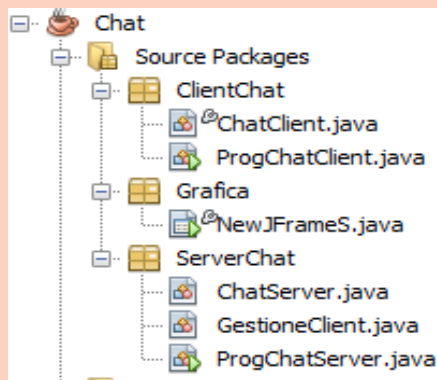
Ho provveduto a lavorare nelle seguenti classi suddivise nei seguenti packages:

Nome del progetto:	Chat
Packages:	1. ClientChat.
	2. ServerChat
	3. Grafica
Classi:	1. ClientChat
	• ChatClient • ProgChatClient
	2. Gafica

- NewJFrameS

### 3. ServerChat

- ChatServer
- GestioneClient
- ProgChatServer



## Fasi Esecutive

1. Dopo la creazione dei vari pacchetti e delle vari classi ho iniziato a programmare le varie richieste partendo dalla classe **ChatServer**.
2. La parte iniziale è il copia incolla del vecchio progetto e quindi si ha la creazione del socket successivamente si ha la sezione del codice che si occupa del nome di riconoscimento di ogni Client che fa anche un controllo in caso di eventuale nome già esistente.
3. Successivamente si avvisa tutti della connessione del Client nella chat
4. Il metodo broadcast serve in una chat Broadcast e nel caso ci fosse un unico Client esso non funziona fino alla connessione di un altro Client.
5. Viene introdotto nel metodo successivo un sistema di avviso nel momento della connessione e della disconnessione dei Client.
6. Successivamente mi sono occupato della creazione della classe **GestioneClient**.
7. In essa vi sono stati creati i vari metodi da successivamente dopo richiamare come il metodo `private void listenForMessages() throws IOException` che mi consente attraverso un controllo di uno switch continuo di usufruire di eventuali comandi nel caso ne avessi bisogno come la disconnessione o la chat singola con un altro Client.
8. Successivamente mi sono spostato nella classe **ChatClient** in essa vi sono i metodi che successivamente ho richiamato nella classe GestioneClient per un maggiore ordine complessivo.
9. Infine mi sono occupato della classe JFrame per introdurre le Swing nel codice.

10. La parte grafica è stata pensata per ora esclusivamente per i Client e quindi ad ogni Client si ha l'apertura di un nuovo Jpanel tutti con una struttura default:



- Parte superiore si ha il titolo della finestra successivamente implementerò il nome del client
- parte centrale si ha la zona della chat e quindi dove essa viene visualizzata e tramite una scrollbar visualizzata.
- Infine nella parte inferiore si ha la possibilità di scrittura e di invio e quindi una conferma della chat.

Il seguente progetto si può trovare nel seguente link: <https://github.com/leoneqafa/Chat-Versione-definitiva>

```
run-single:
```

```
Il server della chat è in funzione, i client si possono connettere alla porta 8888
```

Qafaloku Leone 5 B IA ||| Didattica a distanza

