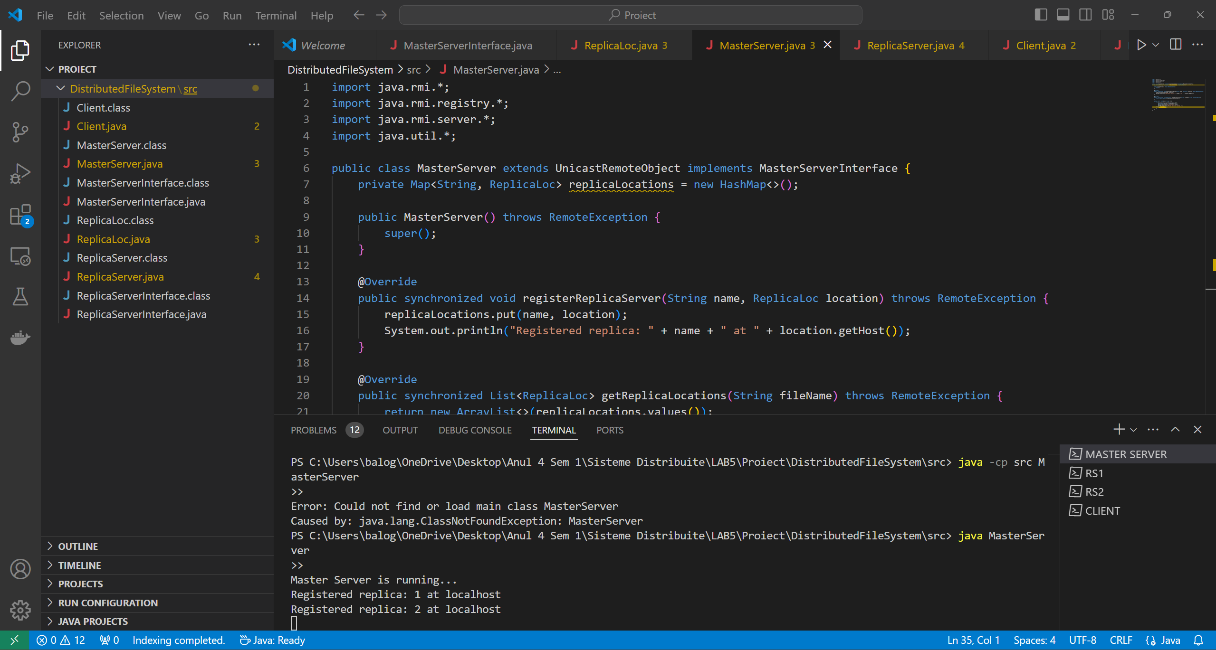
Balog David Alexandru

**Link Github: https://github.com/BalogDavid/L05-SISTEME-DISTRIBUITE**

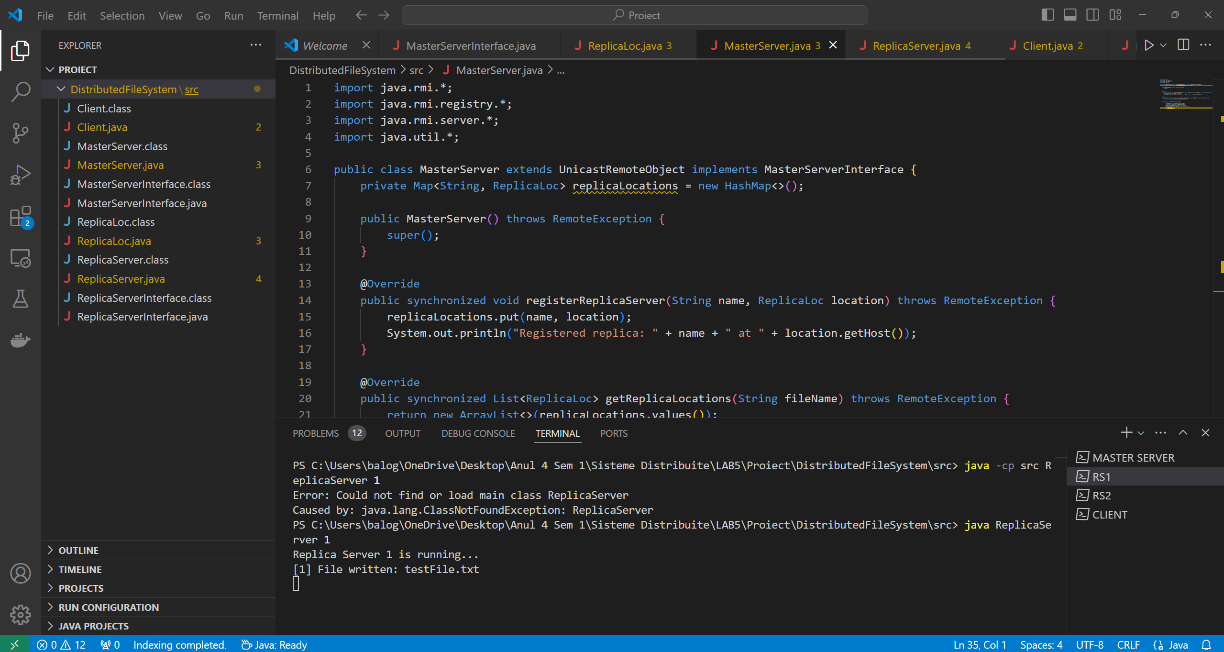
**Arhitectura**

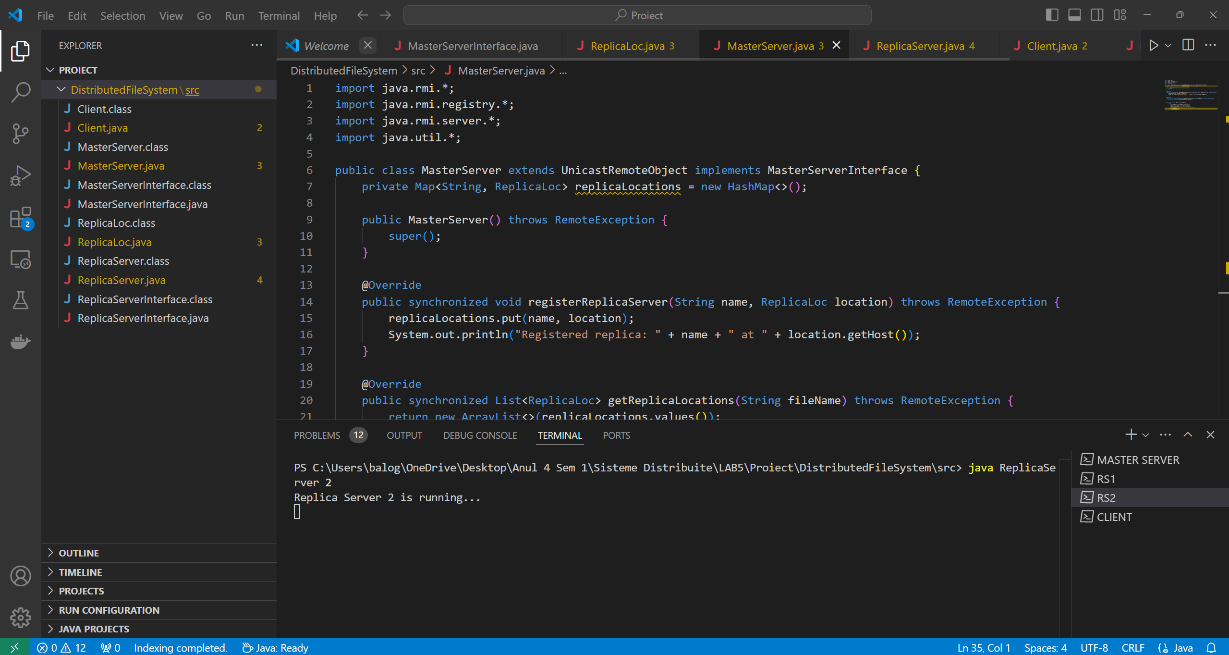
Sistemul de fișiere distribuit implementat utilizează **Remote Method Invocation (RMI)** pentru comunicarea între componente distribuite. Este compus din trei tipuri de noduri:

1. **Master Server**:
   * Rol central care gestionează locațiile serverelor de replicare.
   * Nu stochează date, doar coordonează replicarea și comunicarea între client și replici.

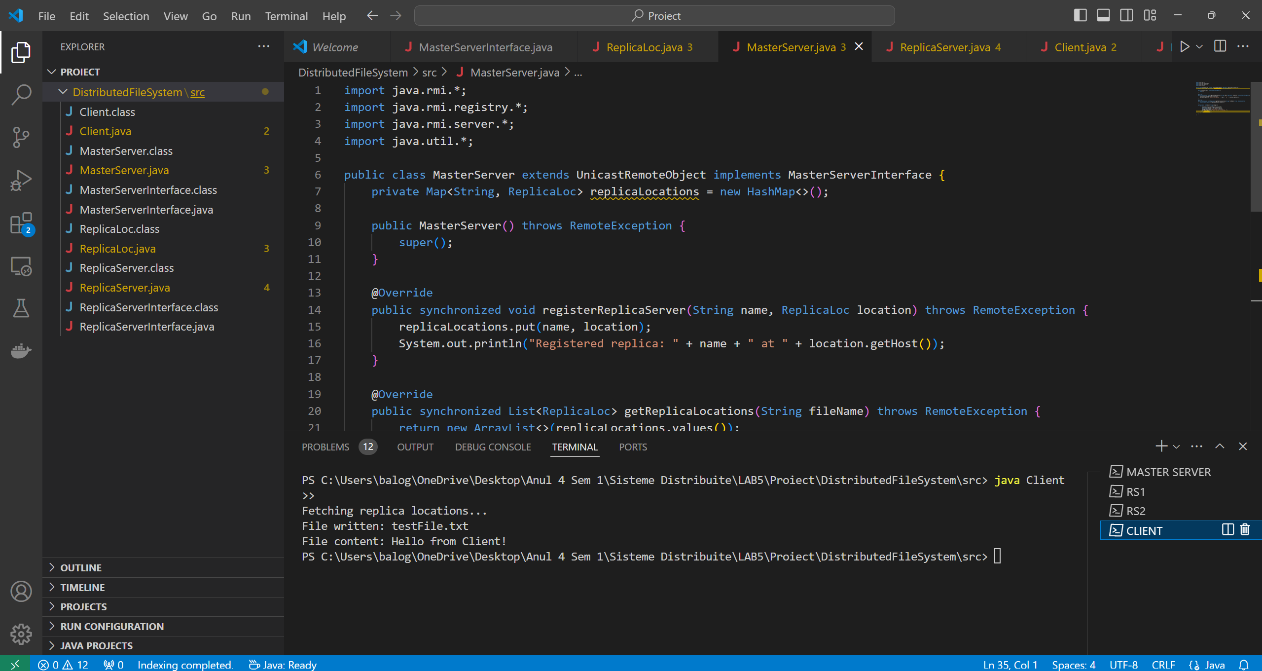


1. **Replica Servers**:
   * Noduri responsabile pentru stocarea efectivă a fișierelor.
   * Organizați într-un lanț utilizând mecanismul de **chained replication**:
     + Scriere: începe de la primul nod și trece prin fiecare nod din lanț.
     + Citire: are loc întotdeauna de pe ultimul nod din lanț, pentru a garanta consistența datelor.





1. **Client**:
   * Aplicație utilizator care trimite cereri de scriere sau citire către serverul master.
   * Se conectează prin RMI la master pentru a primi locațiile replicilor și apoi interacționează direct cu acestea.



**Funcționare**

**Scrierea fișierului**

1. Clientul cere locația serverelor de replicare de la Master Server.
2. Clientul trimite fișierul către primul Replica Server din lanț.
3. Fiecare Replica Server scrie datele local și le transmite mai departe următorului server.
4. Ultimul server finalizează scrierea.

**Citirea fișierului**

1. Clientul cere locația serverelor de replicare de la Master Server.
2. Cererea de citire este redirecționată către ultimul Replica Server din lanț.
3. Replica Server returnează fișierul solicitat clientului.

**Avantaje**

1. **Consistența datelor**:

Citirea de pe ultimul nod din lanț garantează accesul la cele mai recente versiuni ale fișierelor.

1. **Toleranța la defectare**:

Dacă un Replica Server eșuează, restul lanțului poate continua să funcționeze, deși cu performanță redusă.

1. **Distribuirea încărcării**:

Scrierile sunt procesate de mai multe servere, ceea ce reduce riscul de suprasolicitare a unui singur nod.

1. **Extensibilitate**:

Noi servere de replicare pot fi adăugate relativ ușor în lanț.

**Dezavantaje**

1. **Timp de latență crescut**:

Scrierea fișierelor implică propagarea datelor prin toate nodurile din lanț, ceea ce poate deveni lent pentru sisteme mari.

1. **Dependență de noduri intermediare**:

Dacă un nod din mijlocul lanțului eșuează, trebuie repoziționat lanțul pentru a asigura continuitatea.

1. **Single point of failure pentru Master Server**:

Dacă Master Server devine indisponibil, clienții nu mai pot accesa locațiile replicilor.

1. **Lipsa scalabilității pentru citire**:

Toate cererile de citire sunt redirecționate către ultimul nod din lanț, ceea ce poate deveni un punct de îngustare pentru sisteme cu trafic mare.