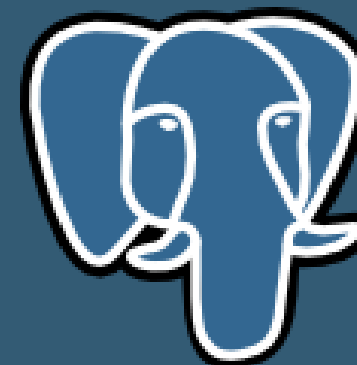




YOUR INFOMOBILITY

NEED



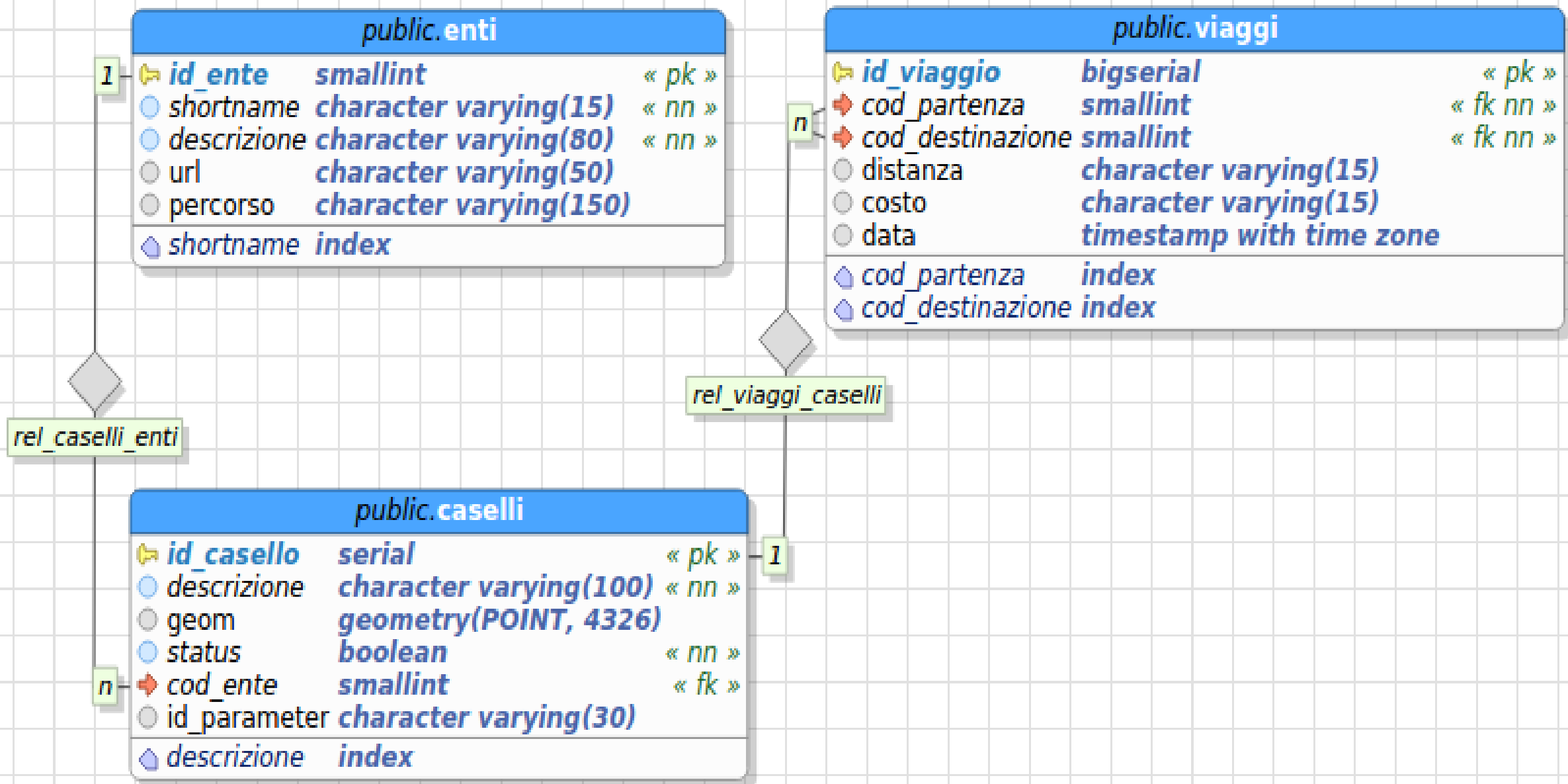
Progettazione

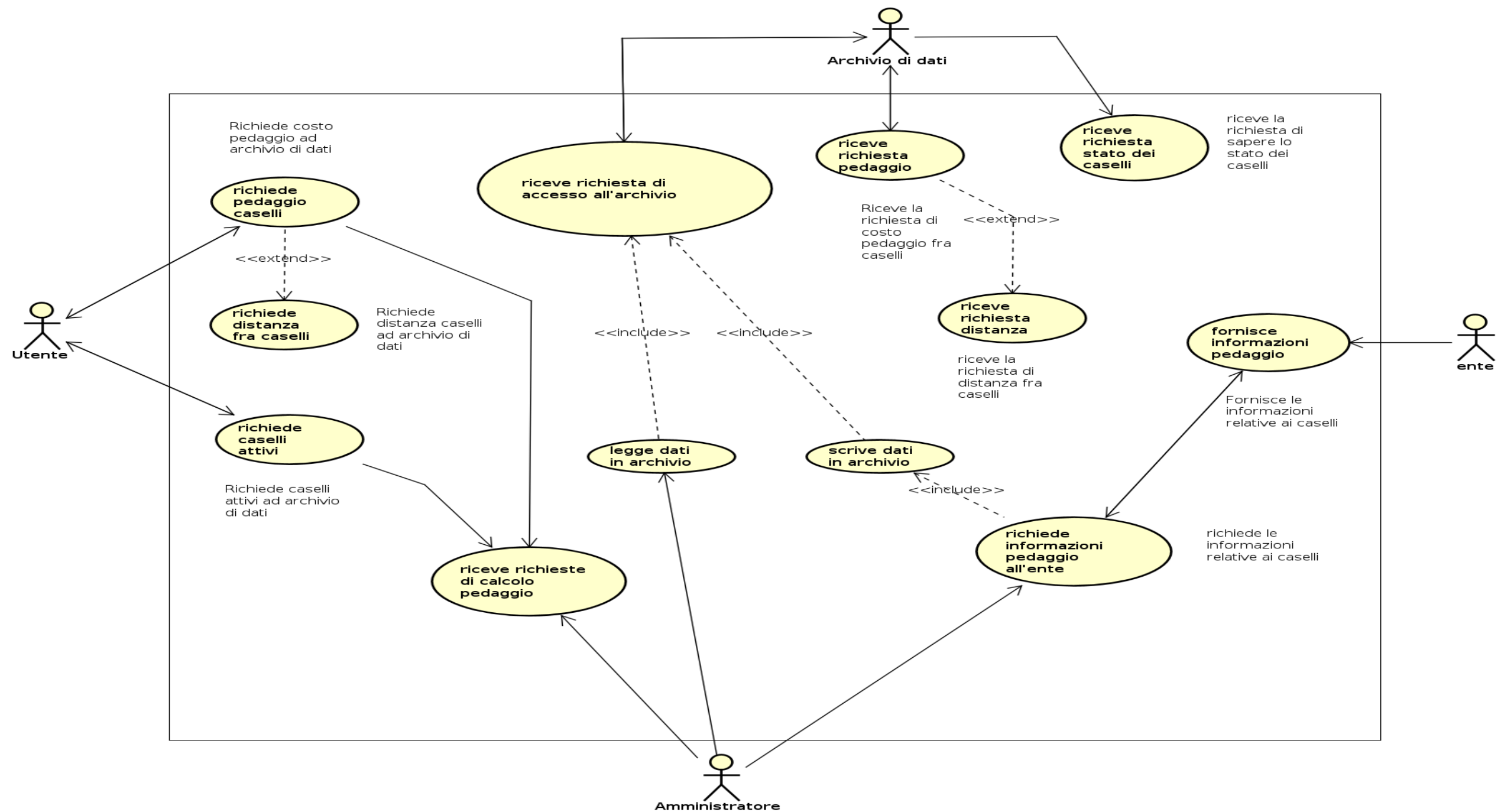
Fase Finale

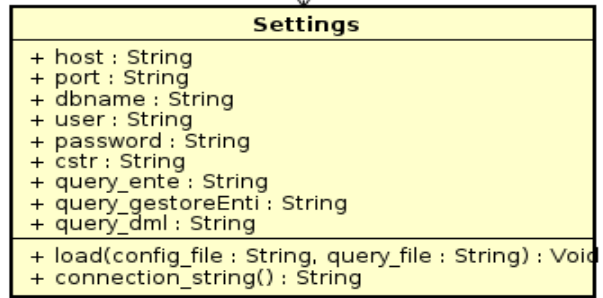
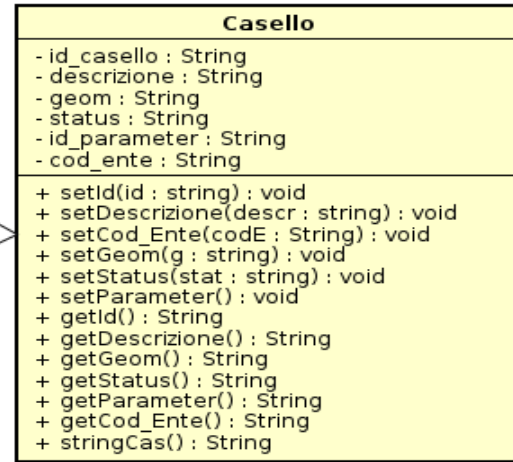
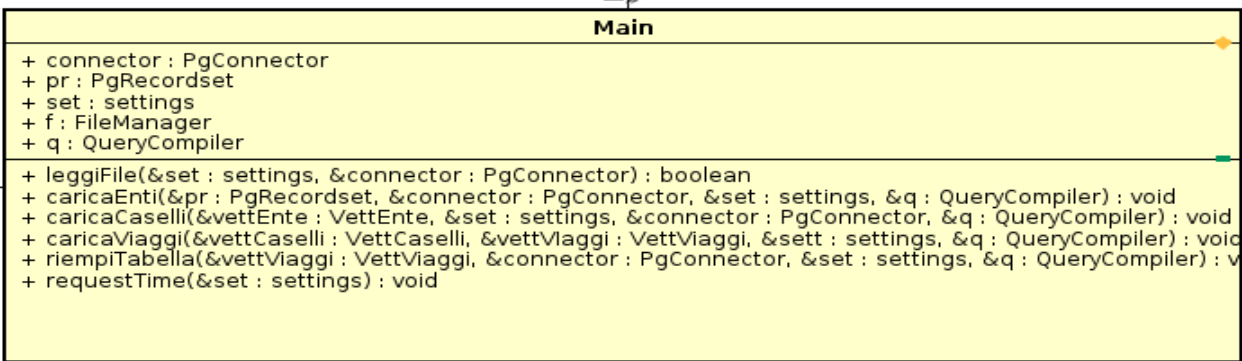
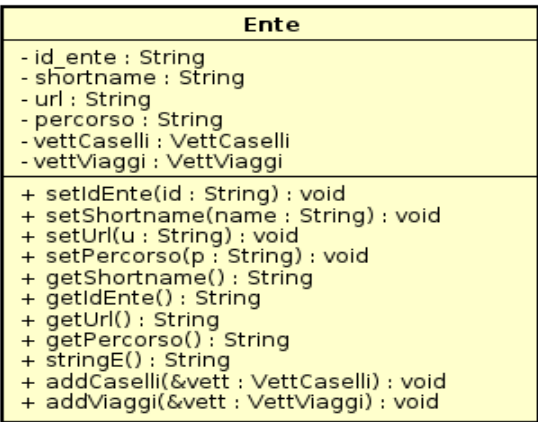
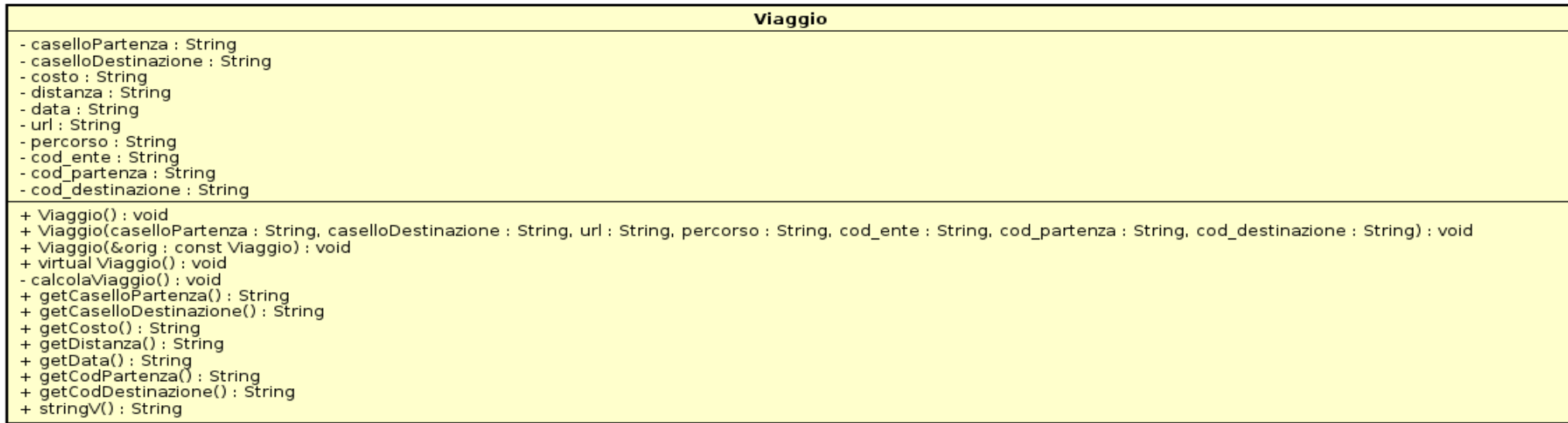
Sviluppo

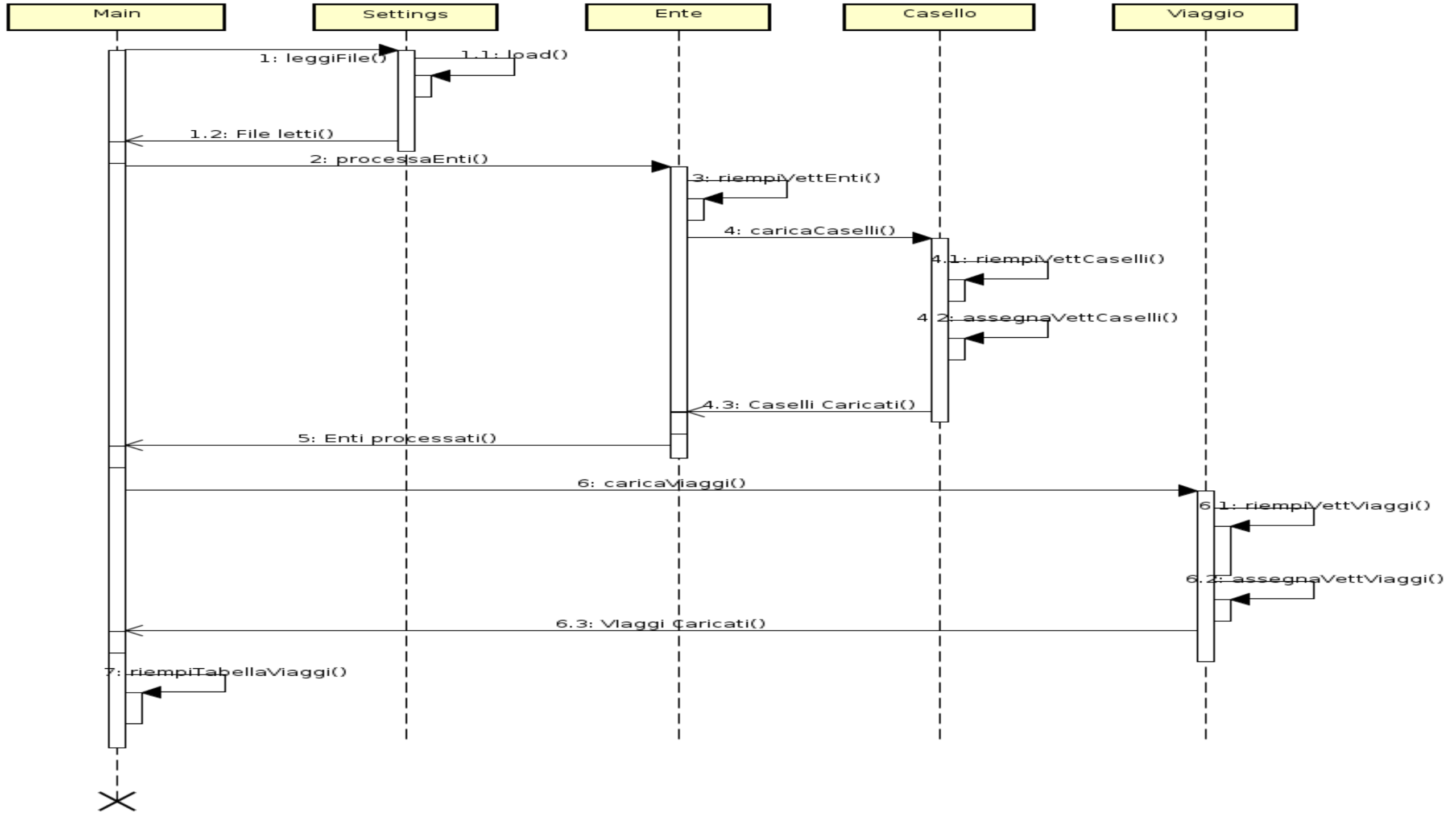


FASE DI
PROGETTAZIONE:
-Francesco Piccolo
-Jia Nian Huang









FASE DI SVILUPPO:

- Daniele Sena
- Marco Borioni

Projects x Start Page x query_database.ini x Main.cpp x

Source History [Icons]

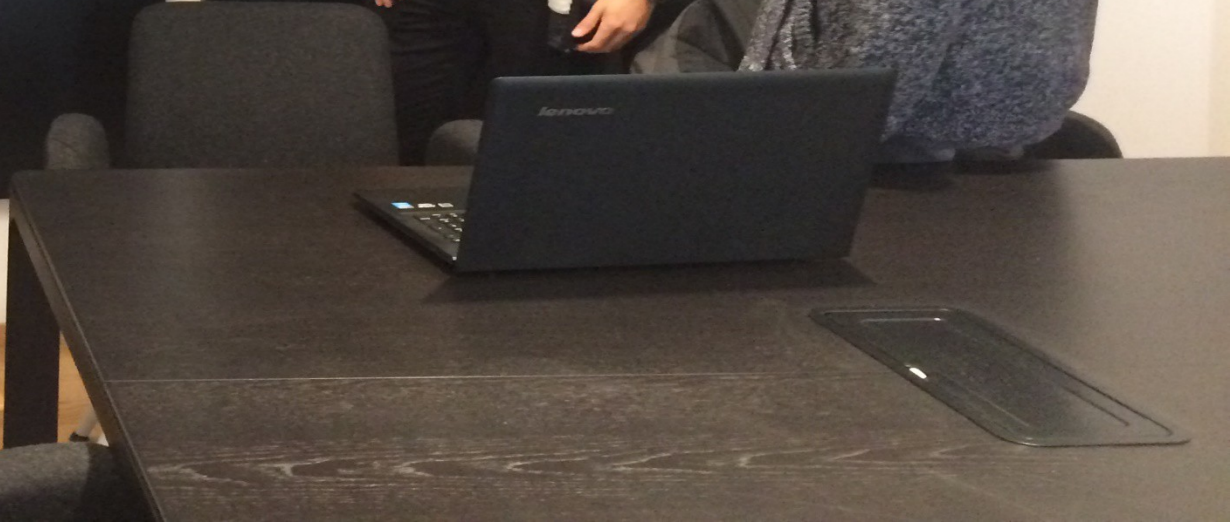
Output Sessions Variables x Call Stack

| Name | Value |
|-------------------|--|
| query_gestoreEnti | "SELECT id_casello, descrizione, geom, status, cod_ente, id_param... |
| vettEnte | {...} |
| [0] | {...} |
| id_ente | "1" |
| shortname | "api" |
| url | "http://www.autostrade.it/autostrade-gis/ricercaPercorso.do?ti... |
| vettCaselli | {...} |
| [0] | {...} |
| id_casello | "" |
| descrizione | "Poggio Imperiale-Lesina" |
| geom | "0101000020E6100000227C6C939DA42E40771C425F98ED4440" |
| status | "t" |
| cod_ente | "1" |
| id_parameter | "3900" |
| [1] | {...} |

Project Structure:

- ProgettoQMAP
 - Header Files
 - Casello.hpp
 - Ente.hpp
 - Settings.hpp
 - Viaggio.hpp
 - iprelibs
 - light_dbms
 - ConnectionHan
 - DatabaseDefine
 - DatabaseExcep
 - PgBulkLoader.h
 - PgConnector.cp
 - PgConnector.hp
 - PgPreparedRec
 - PgPreparedRec
 - PgRecordset.cp
 - PgRecordset.hp
 - QueryCompiler.
 - net
 - http
 - HttpClient.cp
 - HttpClient.hp
 - url_header







FASE DI REVISIONE

FINALE:

-Jose Huertas

-Janluigi De Castro



```
Viaggio(string caselloPartenza, string caselloDestinazione, string url,  
string percorso, string cod_ente, string cod_partenza, string cod_destinazione);
```

```
Viaggio(const Viaggio& orig); //Costruttore che crea un nuovo oggetto e copia gli attributi di un oggetto  
virtual ~Viaggio();
```

```
/*  
 * Solo metodi get, il costruttore ha già settato tutti i dati che servivano  
 */
```

```
string getCaselloPartenza();  
string getCaselloDestinazione();  
string getCosto();  
string getDistanza();  
string getData();  
string getCodPartenza();  
string getCodDestinazione();  
string stringV(); //per una più veloce visualizzazione degli attributi
```

```
private:
```

```
string caselloPartenza = {};  
string caselloDestinazione = {};  
string costo = {};  
string distanza = {};  
string data = {};  
string url = {};  
string percorso = {};  
string cod_ente = {};  
string cod_partenza = {};  
string cod_destinazione = {};  
void calcolaViaggio(); //è privato poiché va a modificare gli attributi costo e distanza
```

```
};  
typedef vector<Viaggio> VettViaggi;  
#endif /* VIAGGIO_HPP */
```

```
string Viaggio::getCodPartenza(){  
    return cod_partenza;  
}
```

```
string Viaggio::getCodDestinazione(){  
    return cod_destinazione;  
}
```

```
/**  
 * PRE-CONDIZIONI: la funzione calcolaViaggio deve essere stata chiamata prima  
 * POST-CONDIZIONI: la stringa deve descrivere solo gli attributi necessari, in questo  
 * caso, quelli da mettere nella tabella viaggi  
 * @return  
 */
```

```
string Viaggio::stringV() {  
    string str = "Casello Partenza: " + caselloPartenza + "\n"  
                "Casello Destinazione: " + caselloDestinazione + "\n"  
                "Costo: " + costo + "\n"  
                "Distanza: " + distanza + "\n"  
                "Data: " + data + "\n"  
                "Cod_Partenza: "+cod_partenza+"\n"  
                "Cod_Destinazione: "+cod_destinazione+"\n";  
    return str;  
}
```



```
for (int i = posDistanza; response[i] != 'K'; i++) {  
    if (response[i] == ',')  
        distanzaPedaggio += '.';  
    else if (isdigit(response[i]) || response[i] == '.')//Isdigit ci consente di ammettere solo  
        distanzaPedaggio += response[i];  
}
```

```
size_t posCosto = response.find("moto:");  
if (posCosto == std::string::npos) {  
    costoPedaggio = "*****Tratto non soggetto a pagamento";  
    break;  
}
```

```
for (int i = posCosto; response[i] != '&' && response[i] != '/'; i++) {  
    if (response[i] == ',')  
        costoPedaggio += '.';  
    else if (isdigit(response[i]) || response[i] == '.')  
        costoPedaggio += response[i];  
}
```

```
}break;
```

