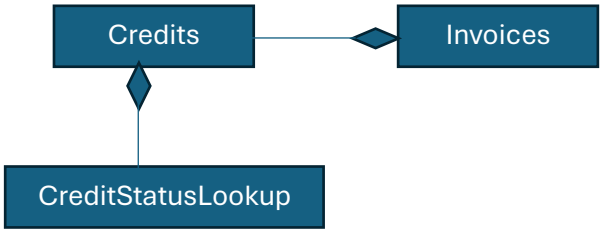
Database Structure



Credits

Stores credits information.

References CreditStatusLookup table.

Script

```
CREATE TABLE IF NOT EXISTS Credits (
    Id INTEGER PRIMARY KEY AUTOINCREMENT,
    Number TEXT NOT NULL,
    ClientName TEXT NOT NULL,
    RequestedAmount DECIMAL(10,5) NOT NULL,
    RequestDate DATE NOT NULL,
    Status INTEGER NOT NULL,

    FOREIGN KEY (Status) REFERENCES CreditStatusLookup(Id)
);
```

Invoices

Stores invoices information.

Has a foreign key to Credits table.

Script

```
CREATE TABLE IF NOT EXISTS Invoices (
    Id INTEGER PRIMARY KEY AUTOINCREMENT,
    Number TEXT NOT NULL,
    Amount DECIMAL(10,5) NOT NULL,
    CreditId INTEGER NOT NULL,

    FOREIGN KEY (CreditId) REFERENCES Credits(Id)
);
```

CreditStatusLookup

Stores the Id and Name of the enum CreditStatus to facilitate comprehension on query results.

Script

```
CREATE TABLE IF NOT EXISTS CreditStatusLookup (
    Id INTEGER PRIMARY KEY,
    Name TEXT NOT NULL
);
```

Architecture Overview

The solution was implemented as a layered architecture using CQRS pattern.

Layers

Presentation

Endpoints for the application.

Project:

DevTask.CreditProcessor.WebApi

The endpoints can be tested using the DevTask.CreditProcessor.WebApi.http file.

Business

Business objects logic of the application.

Projects:

DevTask.CreditProcessor.Domain (business objects and abstractions)

DevTask.CreditProcessor.Application (business logic)

Infrastructure

Data access.

Project:

DevTask.CreditProcessor.Data