

**DME API. Live Search Demo**

Contents

[Introduction 3](#_Toc57179081)

[Installation 3](#_Toc57179082)

[Settings 3](#_Toc57179083)

[Live Search Demo 5](#_Toc57179084)

[Real Time Duplicate Check 7](#_Toc57179085)

[Suppressing Management Schema 10](#_Toc57179086)

[Match Performance Estimate 11](#_Toc57179087)

# Introduction

‘Live Search Demo’ is the example of a standalone application that works with DME API. This example shows how search works in real-time or real-time suppressing management.

'Live Search Demo’ works with Oracle DB, MS SQL Server (or SQL Server Express). At the same time ‘Live Search Demo’ works with DME projects where user controls used database and implemented rules for filtering and suppressing.

There is significant limitation in 1000 records if DME API is used without registration

# Installation

Step 1 - Before start working with ‘Live Search Demo’ application user should check that next applications are installed:

* MS SQL Server or MS SQL Server Express
* Microsoft SQL Management Studio
* .NET Framework (4.5.2 or later)
* Data Match Enterprise

Step 2 - After completing with installation, preparations should be done in database. Create or import database with set of data. Data Match Enterprise can be used for exporting data to SQL Server

Step 3 – After that prepare project in DME. The project must contain only one data source. This data source must be the table in SQL Server. It can be the table prepared during previous step or another table. Save all settings and close DME.

Set some match definitions. For example, how it shows below.

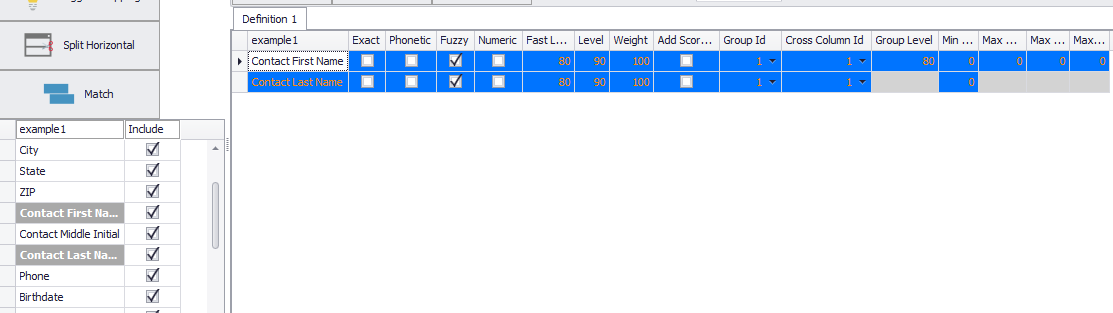


Fig. 1. Example of matching settings in DME

Step 4 - Save project. Close DME.

# Settings

This is WinForms demo application that uses DME API directly, without using of service. The program is intended for searching text in Data Match Enterprise projects. Please note that **only the projects with single data source can be used.**

Application uses configuration.ini and webservice.ini files. Their placement is shown on the Fig.2. Modify these files. Set path settings as it’s defined in Data Match Enterprise. Enter correct connection string These files define where:

* project file search
* search registration key
* keep cached data

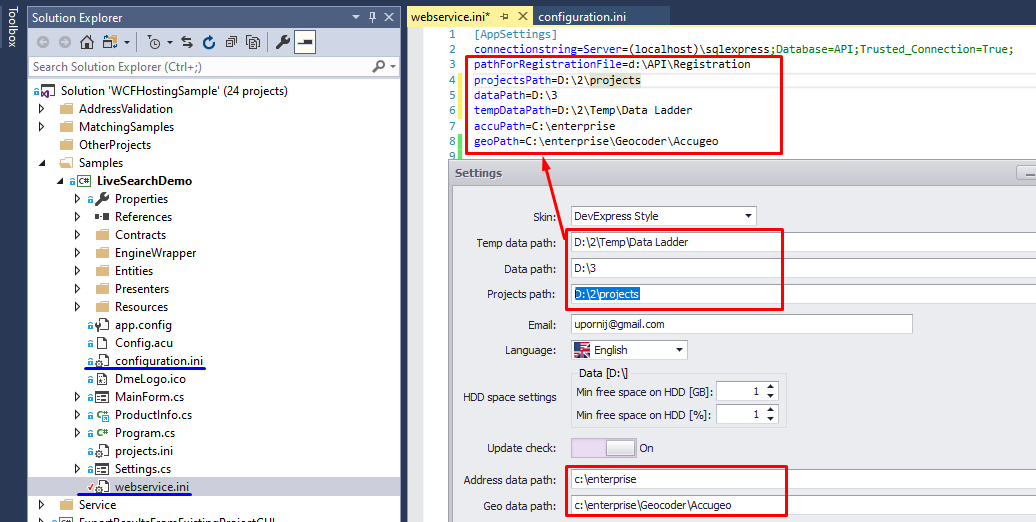


Fig. 2. The project in Visual Studio solution and configurations files

The first file (configuration.ini):

**[AppSettings]**

**pathForRegistrationFile=d:\API\Registration**

**projectsPath=d:\API\Projects**

**dataPath=d:\API\Data**

**tempDataPath=d:\API\Temp**

The second file (webservice.ini):

[AppSettings]

connectionstring=Server=(localhost)\sqlexpress;Database=API;Trusted\_Connection=True;

pathForRegistrationFile=d:\API\Registration

projectsPath=d:\API\Projects

dataPath=d:\API\Data

tempDataPath=d:\API\Temp

accuPath=C:\enterprise

geoPath=C:\enterprise\Geocoder\Accugeo

# Live Search Demo

After configuring please build and start project. If everything configured properly example runs:

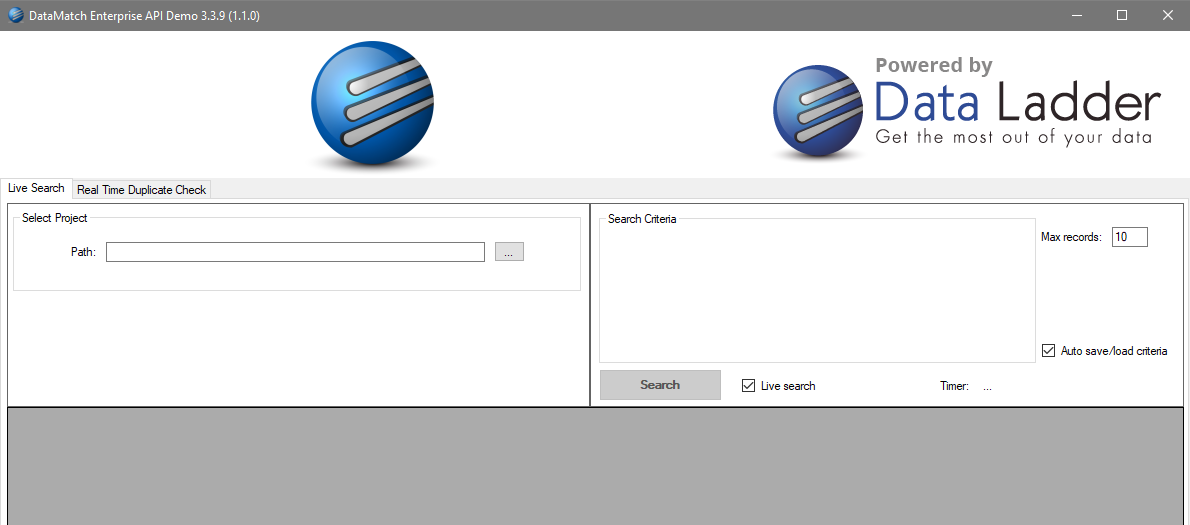


Fig. 1. Main application form appearance

You can prompt project file using standard open dialog clicking on the button …

Then it’s possible to search in matched results using live search or simple search on demand.

When ‘Live Search’ check box is checked then every pressing of key sends an entered data to the Match Engine. Application outputs result of matching into data table on first tab. If check box is not checked, data is sent to result grid when user presses “Search” button.

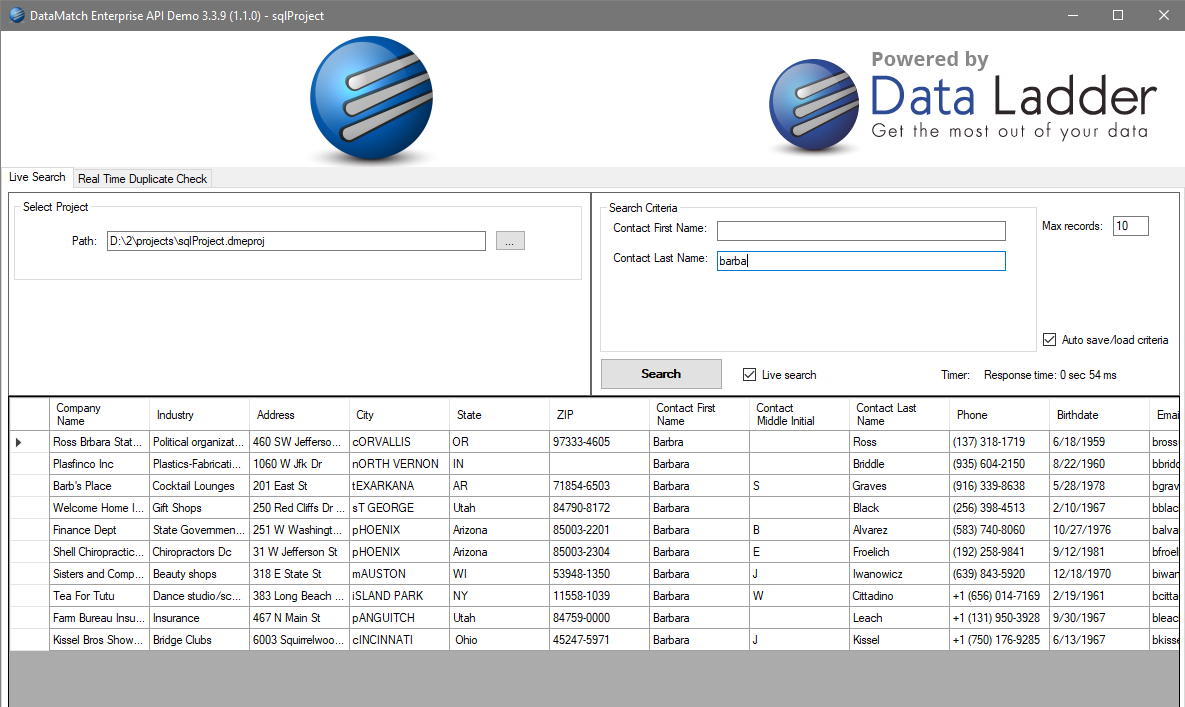


Fig. 2 Live Search Demo application in action

Note: at least 1 match criteria should be defined for search in a project. Textboxes in the top-right corner ‘Contact First Name’ and ‘Contact Last Name’ are corresponded to fields used in match criteria. Entering only Last Name user gets results it’s because the project was chosen with Cross Columns option.

Search results are displayed in the tabular result. Max Record count can be increased for speed of the search process.

# Real Time Duplicate Check

'Real Time Duplicate Check' is the second example that application has. This is the second tab of the application. It checks records before inserting them to the database. If the record is not similar to any existed ones then it will be inserted to the database, otherwise it will be rejected.

The content of table from database is shown in the bottom data table. Duplicates or inserted data are shown in the middle of data table.

’Auto Match’ and ‘Manual Review ‘There are two thresholds that exist in application. If new records have score similarity more than it is set by ‘Auto Match’ then the record will be rejected. If score is less than ‘Manual Review’ it will be inserted. If the score is between thresholds, then manual accepting is required for inserting the record into the database.

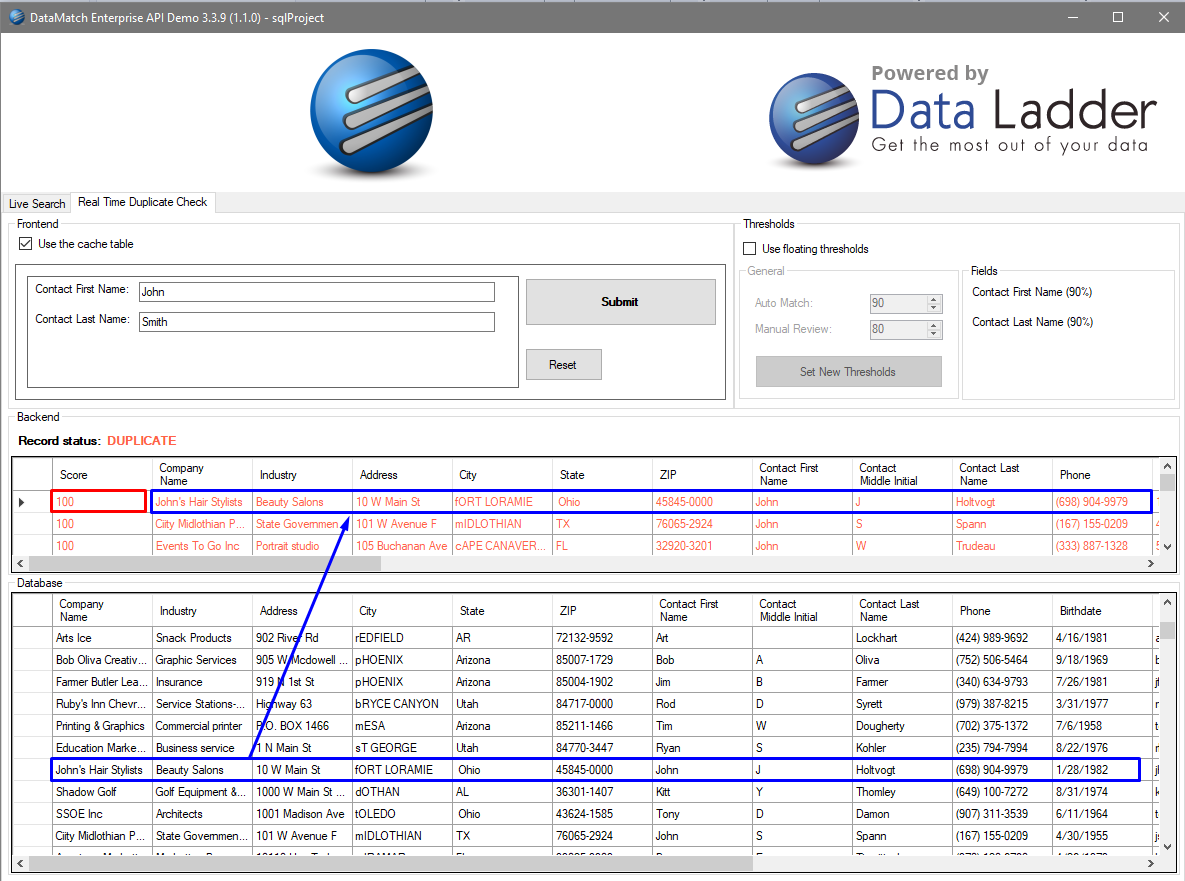


Fig. 3. Real Time Duplication Check application in action (suppressing a duplicate record)

Fig.3 shows rejected insertion. There are a few records with a score equal to 100. It’s bigger than Auto Match level that’s why they were rejected

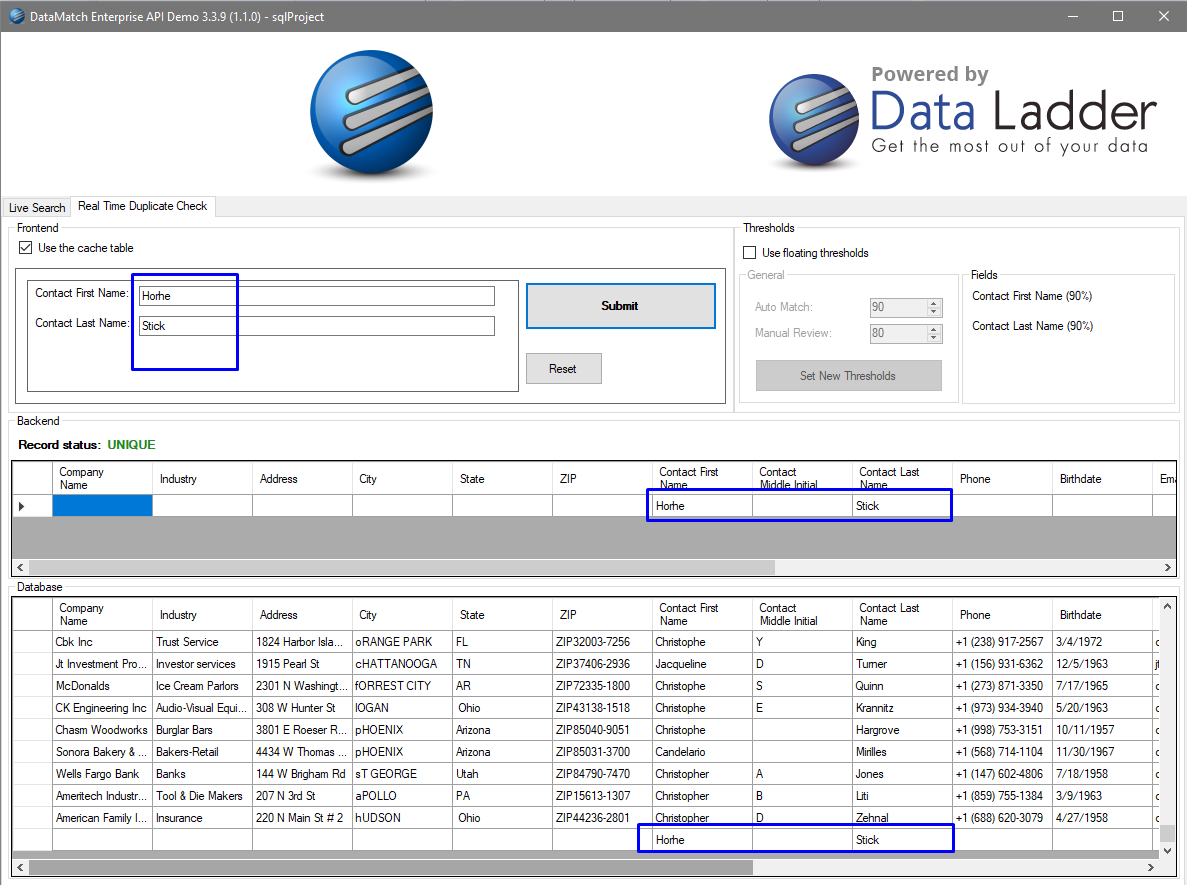


Fig. 4. Real Time Duplication Check application in action (inserting the unique record)

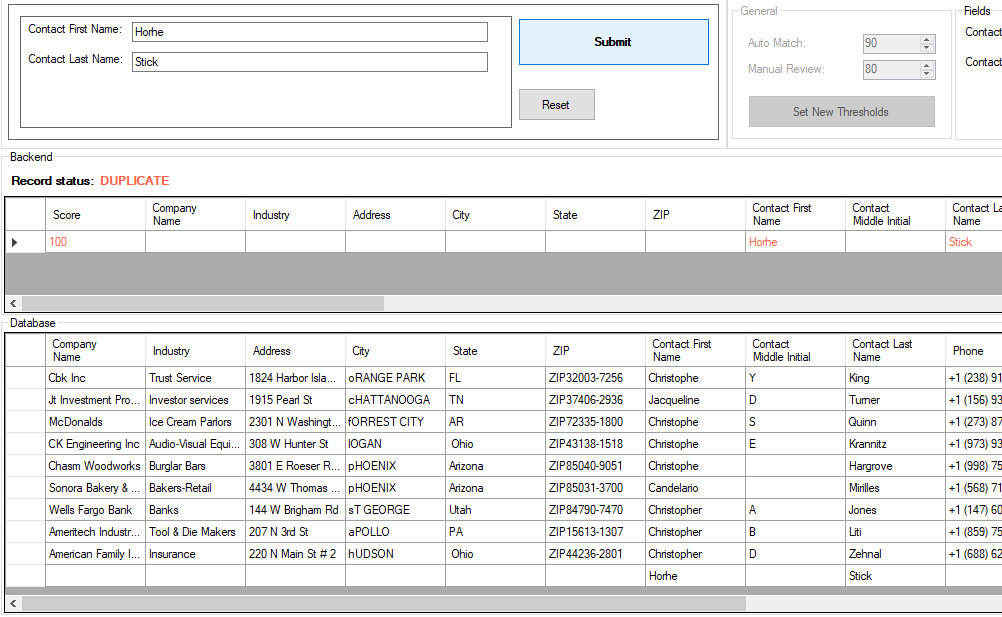


Fig.5 shows how this example works and shows that inserted data can’t be inserted again.

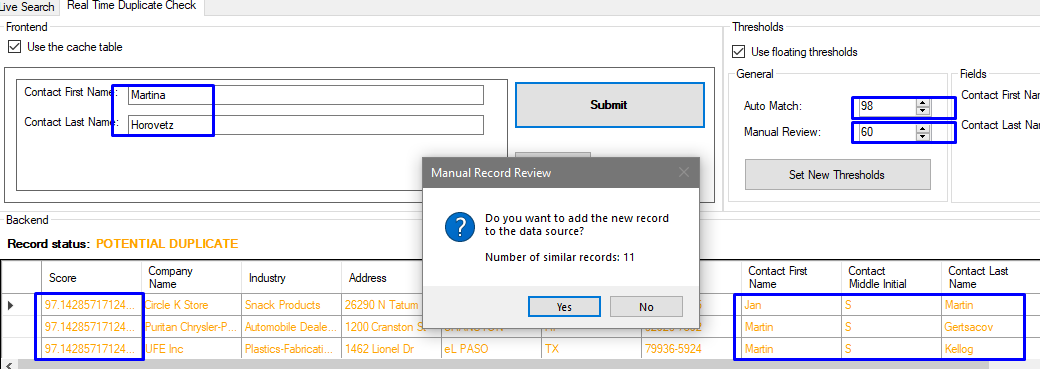
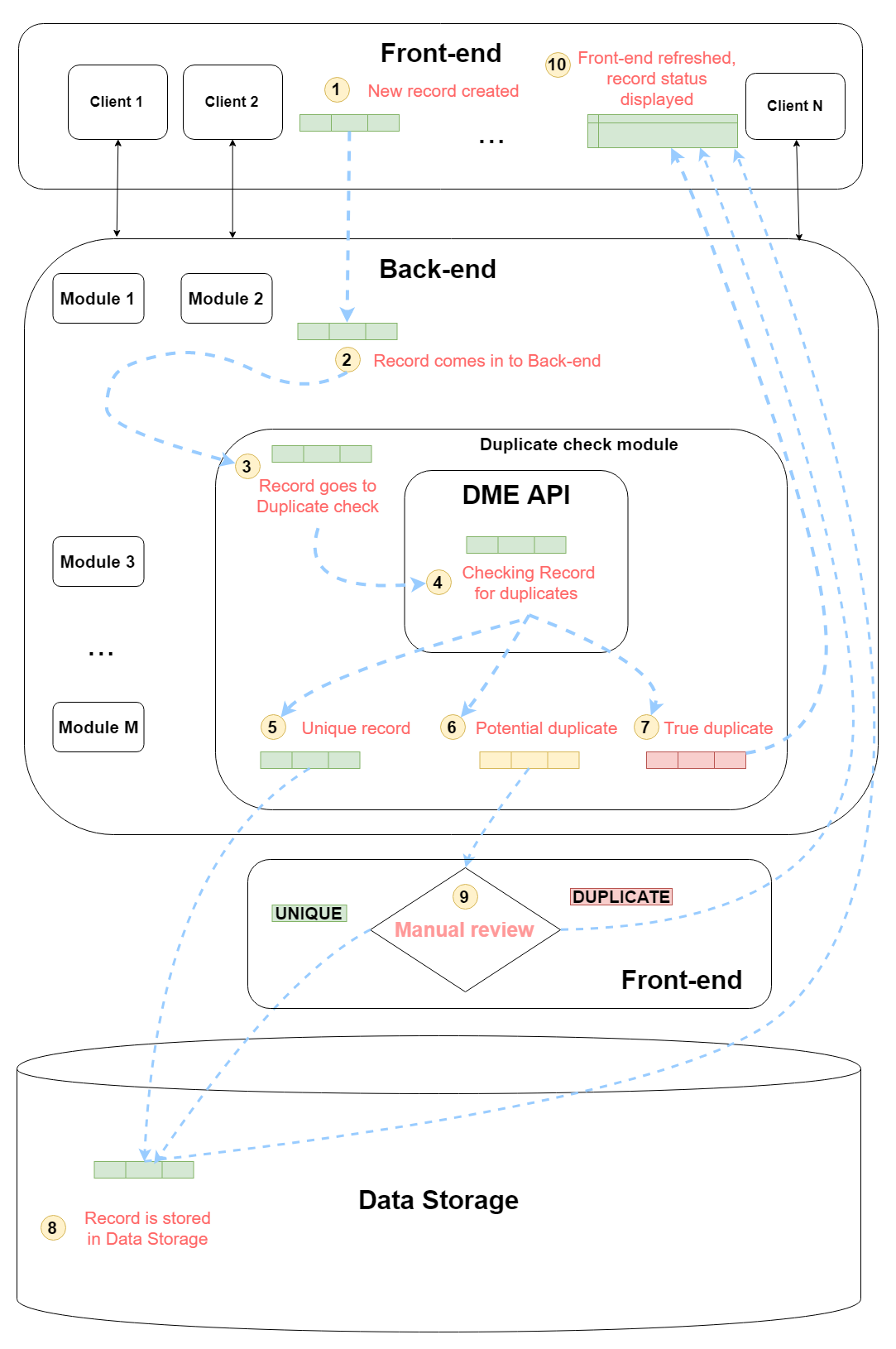


Fig 6 shows inserting the records with score that is less than Auto Level and more than Manual Level

# Suppressing Management Schema

Schema describes all process that are done on Front-End, Back-End and API parts.



# Match Performance Estimate

Approximate estimates were done for ‘Live Search API Demo’ application using PC with 16 GB of RAM and 4 CPUs.

The Matching time depends on a searched word. For some words it is bigger, for others it is less. It depends how many similar words are in a data source.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Data source  (size in GB) | Number of matching fields | Number of records, millions | Load Time, minute | Matching Time, ms | RAM, GB |
| Companies1m.txt (0.19) | 1 | 1 | ~ 1 | 50..60 | 0.6 |
| Companies2m.txt (0.38) | 1 | 2 | ~ 1 | 80..90 | 1.2 |
| Companies2m.txt (0.38) | 2 | 2 | ~ 1.7 | 80..90 | 1.67 |
| 12M 4M dups test.txt (3.6) | 1 | 12 | ~16 | 200…**600**…1000 | 9 |