

S3

Linking Code To Datasets At Labman

Arthur Clarkson

5th September 2024

The main tool we use at Labman to link code to data sets would be Devart's Object Relationship Mapper - Entity Developer. This allows us to develop a database from a model-first-approach, which can update the database structure, and generate C# data models. Using these C# data models, you're able to query data sets with Language Integrated Queries (LINQ) in code, which then gets translated into parametrized SQL to be ran on our server, of which response data is mapped into C# objects.

As an example, for work, I developed this method:

```
/// <summary>
/// Check whether a user with the given id has the given role
/// </summary>
/// <param name="userId">The id of the user to check</param>
/// <param name="roleName">The role to check for</param>
/// <returns>True if user has the given role</returns>
3 references
public bool UserHasPermission(int userId, string roleName) You, 2 months ago • Merged PR 1591: Rename role to permission - Dep...
{
    return _dataContext.Users
        .Where(u =>
            u.Id == userId &&
            u.UserPermissions.Any(a => a.Permission == roleName)
        ).Any();
}
```

It takes a userId and roleName to check if the user has a certain permission, returning true if they do.

This LINQ gets translated into the following SQL:

```
SELECT EXISTS
(
    SELECT t1.*
    FROM users t1
    WHERE (t1.id = :p0) AND (EXISTS
        (
            SELECT t2.id, t2.permission, t2.defaultrole, t2.description
            FROM userpermissions t2
            INNER JOIN userpermissionassignment t3 ON t3.userpermissionid = t2.id
            WHERE (t2.permission = :p1) AND (t1.id = t3.userid)
        )
    )
)
AS C1
-- p0: Input Int (Size = 0; DbType = Int32) [2120]
-- p1: Input VarChar (Size = 25; DbType = AnsiString) [New Starter Board Manager]
-- Context: Devart.Data.MySql.Linq.Provider.MySqlDataProvider Mapping: AttributeMappingSource Build:
5.0.151.0
```

It also allows us to create and update records using code, for example this bit of code finds a bug record, and updates it's durationunits and lastupdatedon fields:

```

[HttpPost] You, 5 months ago • Merged PR 1380: CAN APPROVE NOW - ajaxifying fe...
1 reference
public IActionResult UpdateItemDurationUnits(int bugId, FeedbackDurationUnits units)
{
    // Get the bug and the user.
    var user = _db.Users.FirstOrDefault(u => u.Username == CurrentUsername()) ?? throw new Exception($"User not found. Make sure you are logged in.");
    var bug = _db.Bugs.FirstOrDefault(b => b.Id == bugId) ?? throw new Exception($"Couldn't find a bug or feature with the ID \"{bugId}\"");

    // Check user has permission to change the duration of this item.
    if (!UserCanEditFeedbackItem(bug, user.Id)) throw new Exception("You do not have permission to change this feedback duration");

    _db.Connection.Open();
    _db.Transaction = _db.Connection.BeginTransaction();

    try
    {
        // Update the duration and units of the item.
        if (bug.DurationUnits != units)
        {
            FeedbackDurationUnits oldUnits = bug.DurationUnits;
            bug.DurationUnits = units;
            bug.LastUpdatedOn = DateTime.UtcNow;
            _db.SubmitChanges();

            _feedbackService.CreateCommentForDurationChange(bugId, user.Id, bug.Duration, bug.Duration, oldUnits, units);
            // TODO: potentially send notification to subscribed users?
        }

        _db.Transaction.Commit();
    }
}

```

And this bit of code inserts a bugcomment into the bugcomments table:

```

2 references
public void CreateCommentForDurationChange You, 19 hours ago • Merged PR 1927: Added feedback comments for whe...
(
    int feedbackItemId,
    int userId,
    int? oldDuration,
    int? newDuration,
    FeedbackDurationUnits oldUnit,
    FeedbackDurationUnits newUnit
)
{
    string comment = $"Duration changed from {oldDuration ?? 0} {oldUnit} to {newDuration ?? 0} {newUnit}";
    DateTime date = DateTime.UtcNow;

    Bugcomment feedbackComment = new ()
    {
        BugsId = feedbackItemId,
        UsersId = userId,

        Comment = comment,
        Dateentered = date,

        OldDurationValue = oldDuration,
        NewDurationValue = newDuration,

        OldDurationUnit = oldUnit,
        NewDurationUnit = newUnit
    };

    _db.Bugcomments.InsertOnSubmit(feedbackComment);
    _db.SubmitChanges();
}

```