

System.Transactions

August 17, 2026

System.Transactions

Kind: Service

Source: Pams/Logic/Pams.Business/Pams.Business.csproj (line 1)

Part of: Pams

NuGet package dependency

`System.Transactions` is a NuGet package dependency referenced by `Pams.Business.csproj`. It supports transaction scopes in the Pams.Business layer when backend operations must commit or roll back together.

Diagram

```
sequenceDiagram
    participant Client
    participant API
    participant Business as Pams.Business
    participant Transactions as System.Transactions
    participant Store as Data Store

    Client->>API: Submit operation
    API->>Business: Call business operation
    Business->>Transactions: Open transaction scope
    Business->>Store: Apply data changes
    Store-->>Business: Return result
    Business->>Transactions: Complete transaction
    Business-->>API: Return outcome
    API-->>Client: Return response
```

Usage

```
async function submitOperation(payload: Record<string, unknown>) {
    const response = await fetch("/api/operations", {
        method: "POST",
```

```

    headers: {
      "Content-Type": "application/json",
    },
    body: JSON.stringify(payload),
  });

  if (!response.ok) {
    throw new Error("The operation was not completed.");
  }

  return response.json();
}

await submitOperation({
  accountId: "account-id",
  action: "update",
});

```

AI Coding Instructions

- Treat `System.Transactions` as a backend dependency; TypeScript clients should call the API rather than reference this package.
- Keep related Pams.Business data changes inside the same transaction scope when the operation requires all changes to commit together.
- Do not call `Complete` when validation or persistence fails; disposing an uncompleted scope rolls back its work.
- Check transaction behavior when adding database calls, message publishing, or external service calls inside a business operation.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Injected or called by (1)

- `Pams.Business` — `Pams/Logic/Pams.Business/Pams.Business.csproj` :1