Designing a Workflow Engine Database

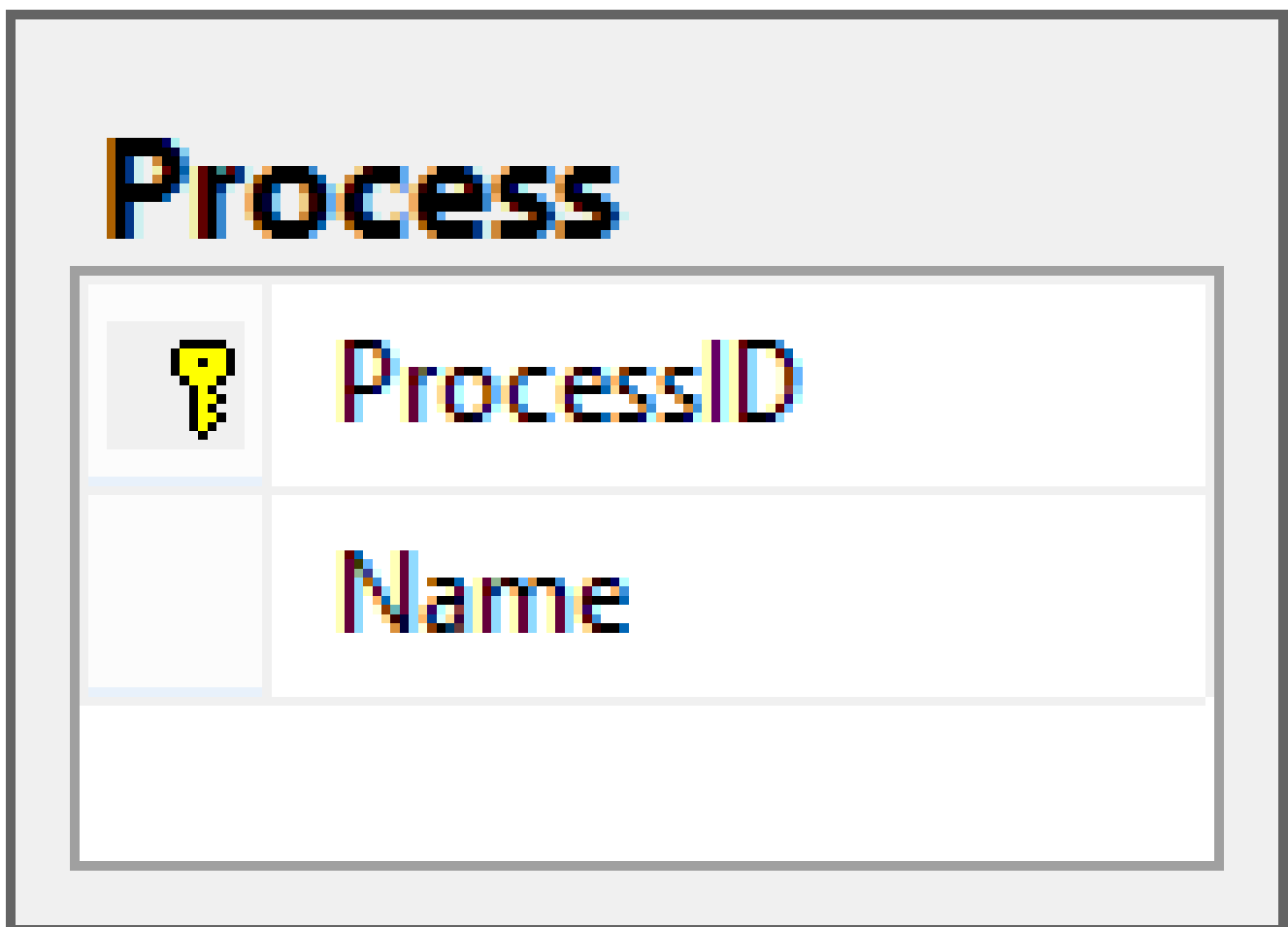exceptionnotfound.net/designing-a-workflow-engine-database-part-2-the-process-table-and-users

This is Part 2 of an eight-part series describing how to design a database for a Workflow Engine. Click here for Part 1

Before we can design anything else in this Workflow Engine Database, we first need to define what exactly makes a Process, as well as who our Users are and which of them can modify the process itself.

The Process

A **Process** is the **collection of all other data that is unique to a group of users and how they want their Requests approved**. In our design, a Process is infrastructure that is used to define and associate most other information in this database.

Our Process table is very simple, as it is just an ID and a Name:

This table, though it is very simple, is the central point of reference for the rest of the design; most of the tables in this database (e.g. States, Transitions, Requests, etc.) will need to be related back to this table, either directly or indirectly.

Users

We also need a table to list the **Users** that can log into this app; this will be a lookup table that has all of our users in it. It'll look like this:
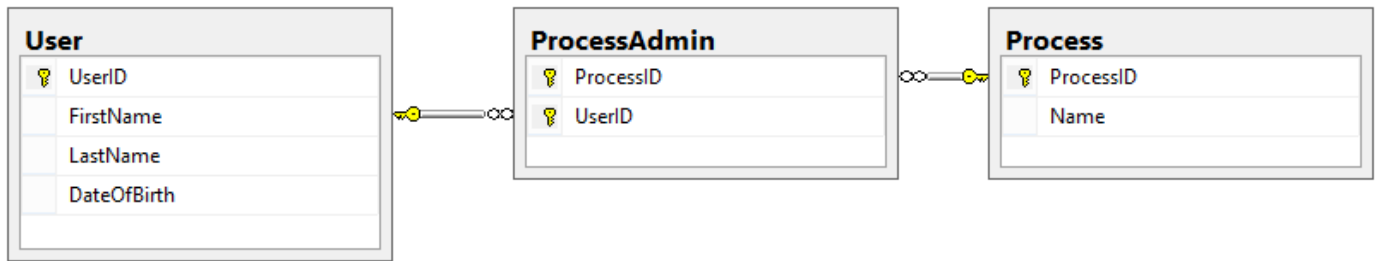

We don't really care how the users get added to this table; we only care that they will exist here and that this table will act as a central access point for user data.

You might be wondering why Users are not specific to Processes. We used this design in order to allow for persons to have jobs in several different Processes without needing to store User data in multiple places.

Process Admins

We also want to allow for a small set of Users that can change the process itself; these people are called **Process Admins**. The table for Process Admins is just a many-to-many relationship table between Process and User:

What did we accomplish?

We created the foundation that the rest of the Workflow Engine can build upon by creating the Process and Users tables, and we established Process Admins that will be able to modify the Process itself.

But this is only the beginning of our journey. In the next installment, we will discuss what actually makes a Request when we talk about **Request Details and Data**.