Redmine - Feature #36933

Halt Redmine boot entirely instead of showing a warning in admin/info when there are pending migrations

2022-04-11 09:11 - Mischa The Evil

Status:	Needs feedback		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Administration		**Estimated time:**	0.00 hour
Target version:	6.1.0			
Resolution:				

Description

Starting with Redmine 5.0.0 admin/info shows a warning when there are pending migrations (#35562). I think it makes no sense to let Redmine boot at all if there are pending migrations. It might even lead to data corruption in the DB.
It might be better to raise an exception and halt the boot process altogether with an initializer (e.g. */config/initializers/05-check_pending_migrations.rb*) containing:

```
if Rails.env.production?
  ActiveRecord::Migration.check_pending!
end
```

What do you think?

History

#1 - 2023-10-29 05:25 - Mischa The Evil

- Status changed from New to Needs feedback

- Target version set to 6.0.0

Given the confirmed potential implications of the issue, I'd would like to propose/discuss this for the upcoming (major) release.

#2 - 2023-10-29 13:05 - Marius BĂLTEANU

I'm not sure if we should entirely halt the boot or we should just throw some warnings in the console.

#3 - 2023-10-30 01:05 - Mischa The Evil

Marius BALTEANU wrote in #note-2:

> I'm not sure if we should entirely halt the boot or we should just throw some warnings in the console.

Let me put it in another way: why should we let Redmine boot into a (production) state that we know upfront, sooner or later will result in errors and/or potential data corruption? Am I missing something? Is there some use-case?

FWIW: I do agree that if we were to let Redmine halt the boot process, we should also provide a clear and meaningful error message to the user.

#4 - 2024-11-03 19:37 - Marius BĂLTEANU

- Target version changed from 6.0.0 to 6.1.0

#5 - 2024-11-04 14:59 - Holger Just

Outright refusing to start entirely with pending migrations may be a bad idea as it may prevent people from running rake tasks which require the environment to load or even just to start a rails console to fix issues with a pending/failed migration.

Instead, if this is in fact desired, we could set config.active_record.migration_error = :page_load in the production environment (it's enabled by default in development). This will insert the
ActiveRecord::Migration::CheckPending:https://api.rubyonrails.org/classes/ActiveRecord/Migration/CheckPending.html rack middleware which will raise a (handled) error if there are any pending migrations when loading a page. Maybe we could even make that configureable in configuration.yml with the setting enabled by default.

It still doesn't feel 100% right, altough I can't really figure out why that is :) But I do think it is actually a good idea to nudge admins to actually run the

migrations. In the end, we might try the middleware approach if that works for "common" upgrades. If there are some missing edge-cases, we may address them as they come up.