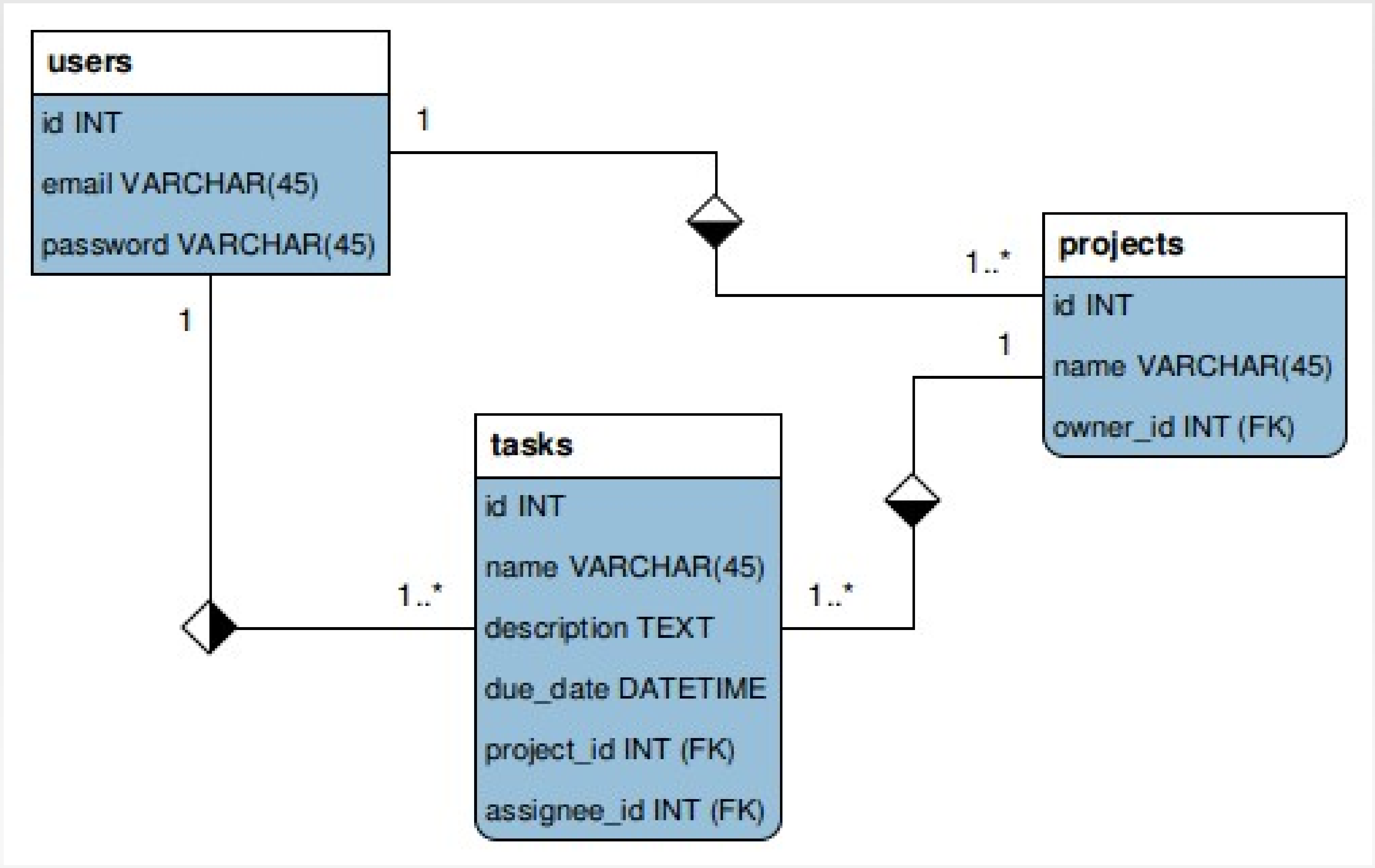
RoR

Create db schema



What gems should I use?

```
What gems should I use?
```

What gems should I use?

Authentication

```
gem 'devise'
```

Authorization

```
gem 'cancancan', '~> 1.7'
```

Simple form:

```
gem 'simple_form'
```

Create users with devise

Run the generator

```
rails generate devise:install
```

Generate User model

```
rails generate devise User
```

Run created migration

```
rake db:migrate
```

Change user factory

```
FactoryGirl.define do
  factory :user do
    email { Faker::Internet.email }
    password { Faker::Internet.password }
  end
end
```

Testing factory

spec/models/user_spec.rb

```
describe User do
  it "has a valid factory" do
    expect(FactoryGirl.build(:user)).to be_valid
  end
end
--------------------------
RAILS_ENV=test rake db:migrate
rspec spec/
```

Create projects

```
Create projects
```

Create projects

```
rails g model Project name:string owner:belongs_to

class Project < ActiveRecord::Base
  belongs_to :owner, class_name: User
end


class User < ActiveRecord::Base
  ...
  has_many :projects, foreign_key: :owner_id
end
```

Create projects

```
[1] pry(main)> FactoryGirl.create :user
=> #<User id: 1, email: "marques@erdman.com", ...>
[2] pry(main)> User.first.projects
  User Load (2.1ms)  SELECT "users".* FROM "users" ORDER BY "users"."id"
ASC LIMIT 1
  Project Load (1.8ms)  SELECT "projects".* FROM "projects" WHERE
"projects"."owner_id" = $1  [["owner_id", 1]]
=> []
[3] pry(main)> User.first.projects.create(name: 'hi')
 => #<Project id: 1, name: "hi", owner_id: 1, created_at: "2014-04-03 01:22:27",
updated_at: "2014-04-03 01:22:27">
```

Create projects

```
[4] pry(main)> Project.first.owner
  Project Load (1.4ms)  SELECT "projects".* FROM "projects" ORDER BY
"projects"."id" ASC LIMIT 1
  User Load (1.9ms)  SELECT "users".* FROM "users" WHERE "users"."id" = $1
LIMIT 1  [["id", 1]]
=> #<User id: 1, email: "marques@erdman.com",...
[5] pry(main)> User.first.projects
  User Load (1.3ms)  SELECT "users".* FROM "users" ORDER BY "users"."id"
ASC LIMIT 1
  Project Load (0.7ms)  SELECT "projects".* FROM "projects" WHERE
"projects"."owner_id" = $1  [["owner_id", 1]]
=> [#<Project id: 1, name: "hi", owner_id: 1, created_at: "2014-04-03 01:22:27",
updated_at: "2014-04-03 01:22:27">]>
```

Create project factory

```ruby
FactoryGirl.define do
  factory :project do
    name { Faker::Lorem.characters(rand(4..30)) }
    association :owner, factory: :user
  end
end
```

Add tests

spec/models/project_spec.rb

```
describe Project do
  it "has a valid factory" do
    expect(FactoryGirl.build(:project)).to be_valid
  end

  it { is_expected belong_to(:owner).class_name(User) }
end
```

Add tests

spec/models/user_spec.rb

```
describe User do
  it "has a valid factory" do
    expect(FactoryGirl.build(:user)).to be_valid
  end
  it { is_expected.to have_many(:projects).
                    with_foreign_key('owner_id') }
end
```

Create tasks

```
Create tasks
```

Create tasks

```
rails g model Task assignee:references project:belongs_to
          due_date:datetime name:string description:text
```

```
class CreateTasks < ActiveRecord::Migration
  def change
    create_table :tasks do |t|
      t.references :assignee, index: true
      t.belongs_to :project, index: true
      t.datetime :due_date
      t.string :name
      t.text :description
      t.timestamps
    end
  end
end
```

Create tasks

```ruby
class Project < ActiveRecord::Base
  has_many :tasks
end


class User < ActiveRecord::Base
  has_many :tasks, foreign_key: :assignee_id
end


class Task < ActiveRecord::Base
  belongs_to :project
  belongs_to :assignee, class_name: User
end
```

Create task factory

```
FactoryGirl.define do
  factory :task do
    association :assignee, factory: :user
    association :project
    due_date { 2.hours.from_now }
    name { Faker::Lorem.words(3).join(' ') }
    description { Faker::Lorem.paragraph }
  end
end
```