**Model**

1- Creating Model named **‘contact\_us’.**

class ContactUs

include ActiveModel::Validations

validates :name, :presence => true

validates :email, :presence => true,

:format => { :with => /\A([^@\s]+)@((?:[-a-z0-9]+\.)+[a-z]{2,})\Z/i, :message => "email is invalid" }

validates :description, :presence => true,

:length => {:minimum => 10, :maximum => 300}

# to deal with form, you must have an id attribute

attr\_accessor :id, :name, :email, :description

def initialize(attributes = {})

attributes.each do |key, value|

self.send("#{key}=", value)

end

@attributes = attributes

end

def read\_attribute\_for\_validation(key)

@attributes[key]

end

def to\_key;

end

def save

if self.valid?

Emails.contact\_us(self).deliver

return true

end

return false

end

def persisted?

false

end

end

**Controller**

1- Adding **‘contact’** and **‘contact\_us\_create’** method in the StaticPagesController.

class StaticPagesController < ApplicationController

**def contact**

**@contact\_us = ContactUs.new(:id => 1)**

**end**

**def contact\_us\_create**

**@contact\_us = ContactUs.new(params[:contact\_us])**

**if @contact\_us.save**

**flash[:notice] = "Thanks for getting in touch! You will receive a response from us within 24 hours. Have a wonderful day ahead!"**

**redirect\_to :back**

**else**

**flash[:error] = "Could not send email."**

**render 'contact'**

**end**

**end**

end

**Views**

**1-** Adding **‘contact us form’** in the view file located in the **‘views/static\_pages/contact.html.haml’**

**%h3**

**Send a Message**

**%br**

**= simple\_form\_for @contact\_us, :url => { :action => "contact\_us\_create" }, :validate => true do |f|**

**= f.input :name, :label => 'Your name please'**

**= f.input :email, :placeholder => 'user@domain.com'**

**= f.input :description, :as => 'text'**

**= f.button :submit, :class => ‘btn btn-primary’**

**Routes**

**1-** Adding ‘contact’ and ‘contact\_us\_create’ routes to the routes.rb file.

**match 'contact' => 'static\_pages#contact'**

**match 'contact\_us\_create' => 'static\_pages#contact\_us\_create'**

**Mailer**

1- Inside ‘app/mailers’ make one file named **emails.rb** and define **‘contact\_us(sender)’** function:

**class Emails < ActionMailer::Base**

**def contact\_us(sender)**

**@sender = sender**

**mail(:to => 'vinti.uiet@gmail.com',**

**:from => sender.email,**

**:subject => "[RailsforCharity]: Email from " + @sender.name)**

**end**

**end**

**Config**

1- Inside **‘config/environments/development’** change action\_mailer delivery\_errors from false to true.

**config.action\_mailer.raise\_delivery\_errors = true**

**2-** Inside **‘config/initializers’** make file **init\_action\_mailer.rb:**

**ActionMailer::Base.delivery\_method = :smtp**

**ActionMailer::Base.perform\_deliveries = true**

**ActionMailer::Base.default :charset => "utf-8"**

**ActionMailer::Base.smtp\_settings = {**

**:address => ENV['MAIL\_ADDRESS'],**

**:port => ENV['MAIL\_PORT'],**

**:domain => ENV['MAIL\_DOMAIN'],**

**:user\_name => ENV['MAIL\_USERNAME'],**

**:password => ENV['MAIL\_PASSWORD'],**

**:authentication => :plain,**

**:enable\_starttls\_auto => true**

**}**

**3-** ENV variables can be define inside home folder:

$ cd ~/.private\_settings

$ mkdir railsforcharity

$ cd railsforcharity

$ mate development

**# Gmail**

**export MAIL\_ADDRESS=smtp.gmail.com**

**export MAIL\_PORT=587**

**export MAIL\_DOMAIN=www.gmail.com**

**export MAIL\_USERNAME=info@example.com**

**export MAIL\_PASSWORD=xxxxxxxxxx**

**4-** As I defined all variables in the environment, that’s why I need to run that **~/.private\_settings/railsforcharity/development** file in the bash shell. And to run it on one time basis, you can copy the all content of that file and paste it in the terminal and then run rails s, then It will automatically take all variables

And, later we need to make a rake task to run that file.

**Additional Features**

**1- To add Google map:**

just open maps.google.com, update required address and click on the **link** button at the top, you will find find **iframe tag inside ‘Paste HTML to embed in website’ heading**

like this:

**<iframe width="425" height="350" frameborder="0" scrolling="no" marginheight="0" marginwidth="0" src="https://maps.google.com/?ie=UTF8&amp;ll=37.397161,-122.024121&amp;spn=0.152741,0.307274&amp;t=m&amp;z=12&amp;output=embed"></iframe><br /><small><a href="https://maps.google.com/?ie=UTF8&amp;ll=37.397161,-122.024121&amp;spn=0.152741,0.307274&amp;t=m&amp;z=12&amp;source=embed" style="color:#0000FF;text-align:left">View Larger Map</a></small>**

copy this and paste in the view file(contact.html.haml) where you want to show map.

**2- simple\_form gem:**

I used simple\_form gem to make form, but while defining input field like:

**f.input :description, :as => ‘text’,** the box appeared is very small so to make it bigger, i did this:

**f.input :description, :as => ‘text’, :input\_html => {:style => ‘:width: 600px’, :rows => 10}**