

## Models:

app/models/application\_record.rb

```
<<Abstract>>  
ApplicationRecord < ActiveRecord::Base
```

[ActiveRecord](#)

app/models/url\_entry.rb

```
class UrlEntry < ApplicationRecord
```

id: Integer  
url: String  
description: String  
expire: DateTime  
created\_at: DateTime  
updated\_at: DateTime  
counter: Integer  
archived: Boolean  
sponsored: Boolean

Public:  
active(): ActiveRecord\_Relation  
sponsored\_first(): ActiveRecord\_Relation  
increment(): Boolean

app/models/setting.rb

```
class Setting < ApplicationRecord
```

id: Integer  
filter\_chars: String  
created\_at: DateTime  
updated\_at: DateTime

Public:  
filter\_chars\_array(): Array

## Controllers:

app/controllers/application\_controller.rb

```
class ApplicationController < ActionController::Base
```

[ActionController](#)

request : [ActionDispatch::Request](#)

response : [ActionDispatch::Response](#)

app/controllers/url\_entries\_controller.rb

```
class UrlEntriesController < ApplicationController
```

@url\_entries: UrlEntry::ActiveRecord\_Relation

@url\_entry: UrlEntry

Public:

index(request): response

show(request): response

search(request): response

visit(request): response

new(request): response

edit(request): response

create(request): response

update(request): response

destroy(request): response

Private:

settings\_filter\_chars()

set\_url\_entry()

url\_entry\_params()

valid\_url?()

perform\_or\_search()

perform\_and\_search()

perform\_not\_search()

app/controllers/settings\_controller.rb

```
class SettingsController < ApplicationController
```

@settings: Setting::ActiveRecord\_Relation

@setting: Setting

Public:

index(request): response

show(request): response

update(request): response

purge(request): response

Private:

setting\_params()

set\_setting()