**Projectbox**

An open source project-sharing application for the World Wide Web and mobile systems

Course: Software Engineering Project II (CICS 530)

Term: Winter-2, 2012

Submission date: 23rd April, 2012

Instructor: George Tsiknis

TA: Ricky Cheng

**Group Members:**

Matthew Robertson

Gayathiri Charathsandran

Ying Xu (Lillian)

Srikanth Ramu

Benjamin Bertka

Parvez Islam

Master of Software Systems (MSS)

Institute for Computing, Information and Cognitive Systems (ICICS)

University of British Columbia (UBC)Table of content

# 1.0 Summary

*Projectbox* is a simple collaboration tool meant for both web and mobile platforms. It is easy to use and a user can have a project uploaded, shared and available for collaboration in minutes.

# 2.0 Glossary

1. Projectbox: This is the name of our system. The system can be used for hosting, sharing and contributing Open Source projects. There are contributors who help in the technical aspects of a project while project managers are the owner of a project
2. Web Platform: This pertains to the web interface which can be accessed using a browser from any desktop and laptop computer.
3. Mobile Platform: This pertains to the mobile interface which is meant for devices with limited computing resources such as iPhone, Android phones, Blackberry, etc.
4. RoR/Rails: Ruby on Rails MVC framework.
5. Project manager: A project manager in the context of our Projectbox system is the owner/creator of a project. A project manager may add/delete contributors to/from a project.
6. Contributor: A contributor is created when either invited to join a project or the project manager accepts a user’s request to join a project.
7. Administrator: The administrator of Projectbox has access to all projects and user information. An administrator reserves the right to create/modify/delete a project/user.

**3.0 Introduction**

*Projectbox* is a simple collaboration tool meant for both web and mobile platforms. It is an interactive and easy to use tool where user can have a project uploaded, shared and available for collaboration in minutes from anywhere. Project upload can be anything files, images or video clips to the project. It was built using Ruby on Rails MVC architecture and phone gap mobile framework. It is compatible with Android smart phones to access from anywhere.

This powerful tool can be used for collaborating professionals in a frame work, ease their communication among projects and easily handle separation between personnel’s owned projects and contributing projects. It is an easy tool for Project Managers to create and manage their projects. This tool can be used to recruit project contributors via email or can ask the existing ProjectBox users to contribute. For each user of this tool, who interacts either by PC or smart phone is provided with appropriate interface to access the ProjectBox.

Administrators of this ProjectBox can maintain the overall system and send announcement for users. Project managers can create, update and delete their own project and can decide whom to join the project.

Below is a list of our intended usage:

Organization who wanted to

* maintain their project management activities or
* provide an easy tool for their project managers and professionals to perform their task effective from anywhere or
* run a business to provide the service

Users of ProjectBox,

* Project managers
* Professional who wanted to contribute to the project
* Users of the projects resources created by project managers

**4.0 System Architecture**

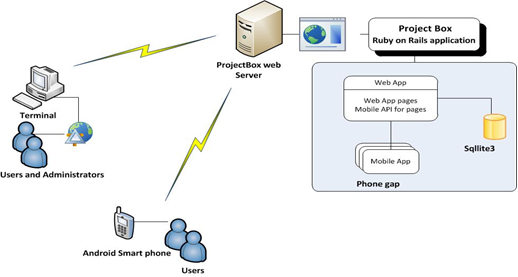


Figure: Interactions between project box servers and clients (desktop and mobile). Mobile architecture discussed more later.

**5.0 System requirements**

Administrator:

* Rails server (Open source and can be easily downloaded)
* Phone gap (Open source and can be easily downloaded)
* Domain name in the Internet

Project Managers / project contributors:

* Internet access
* Registration with service providers

**6.0 Performance Requirements**

This section specifies non-functional requirements such as:

* Response times: we would expect the system to respond to each request in 1 second
* Capacities (number of simultaneous users, database sizes, etc.): We have deployed the system on Amazon AWS servers which provides distributed services. This means our system can handle infinite number of requests simultaneously and the user database can grow infinitely.
* Reliability and availability: We have built the system with both reliability and availability in mind. The RoR system itself provides several options for this.
* Security: Rails as an MVC system provides built in security such as password encryption-algorithms, seeding and standard sever side security.
* Usability: We have successfully managed to design the system with high usability in mind. We have used twitter-bootstrap code to simplify the user experience. The theme has been incorporated from several popular websites (just the idea).

**7.0 Use Cases**

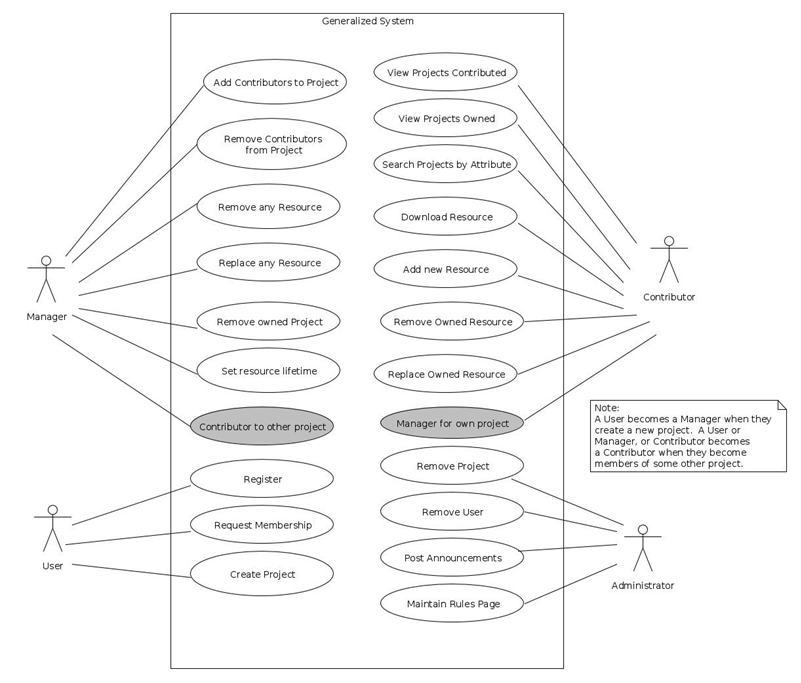
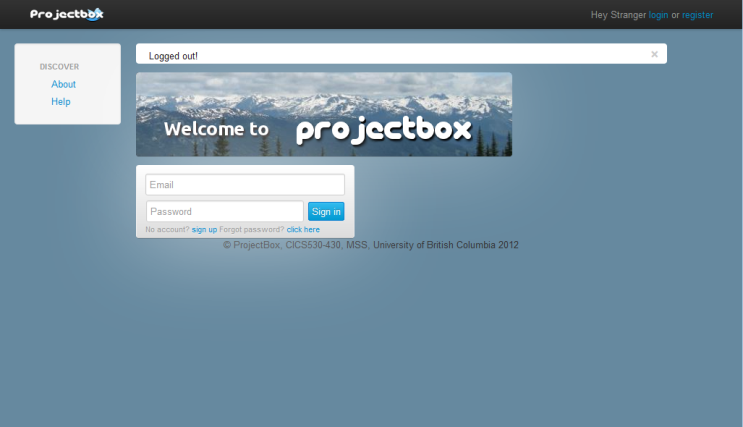
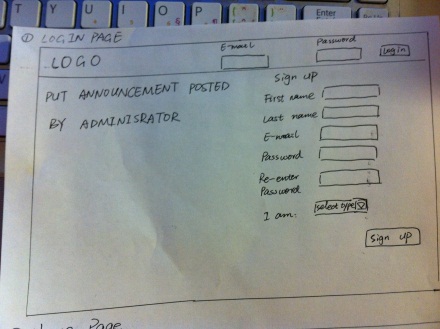


Figure: Generalised use case diagram

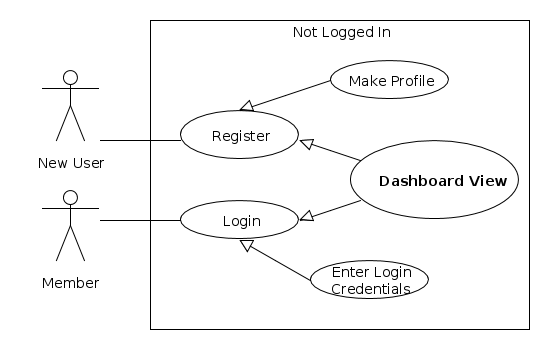
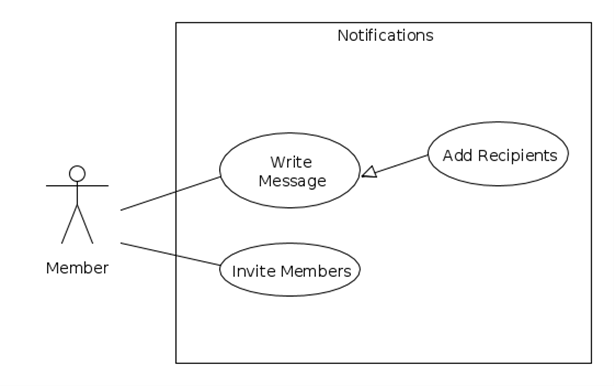
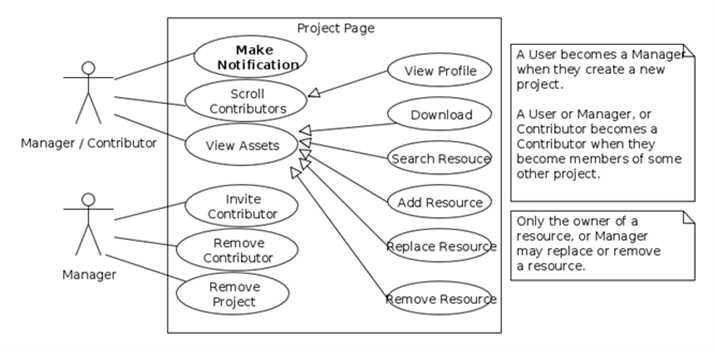
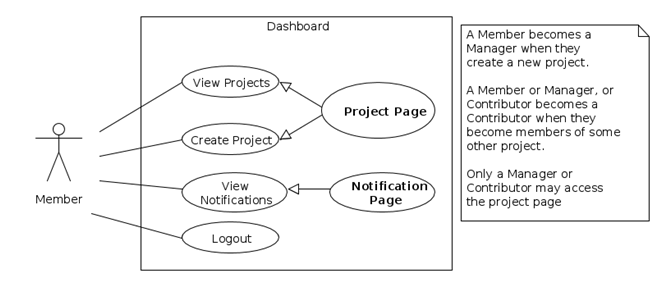
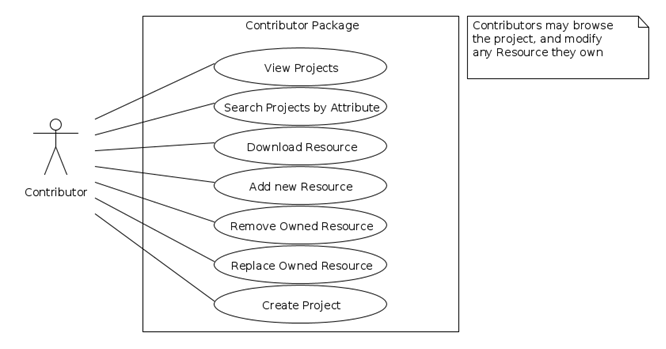
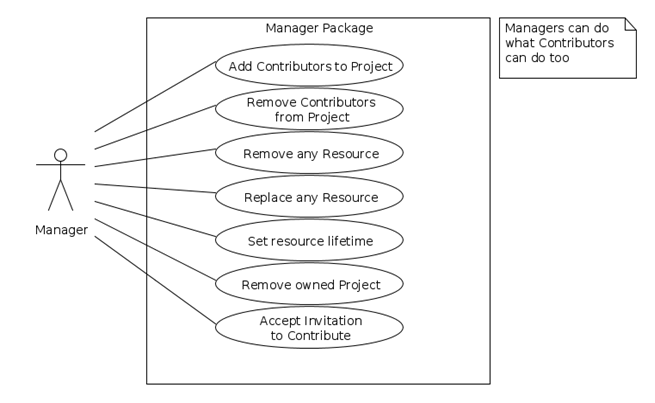
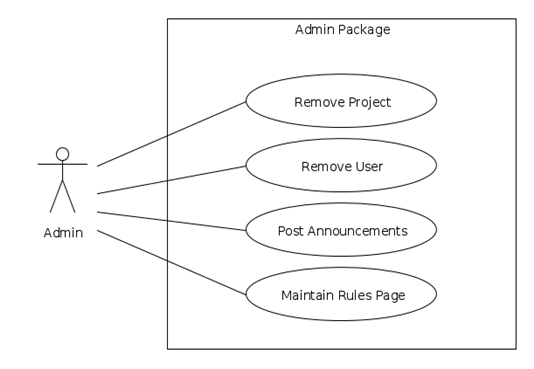
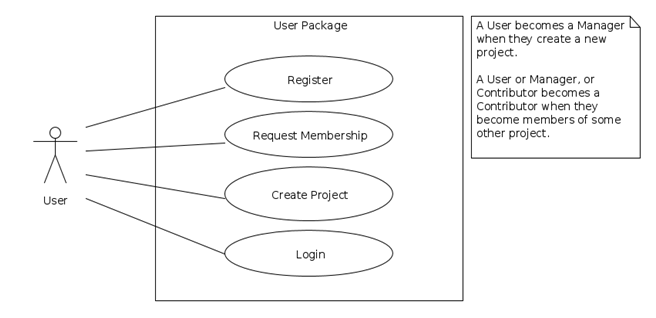
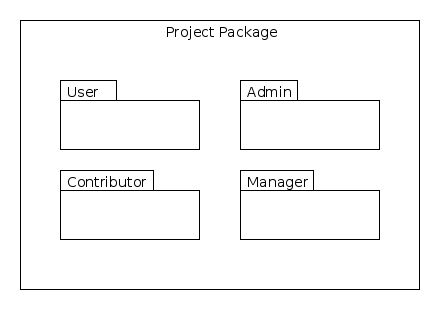
# 

# 8.0 User Interface Design

We have used the Pencil tool to draw UI mock-ups before we actually deployed the design. Below are some of the pre-sketches, pencil diagram and final look:



# 9.0 Behaviour Diagrams



# 10.0 Database Design

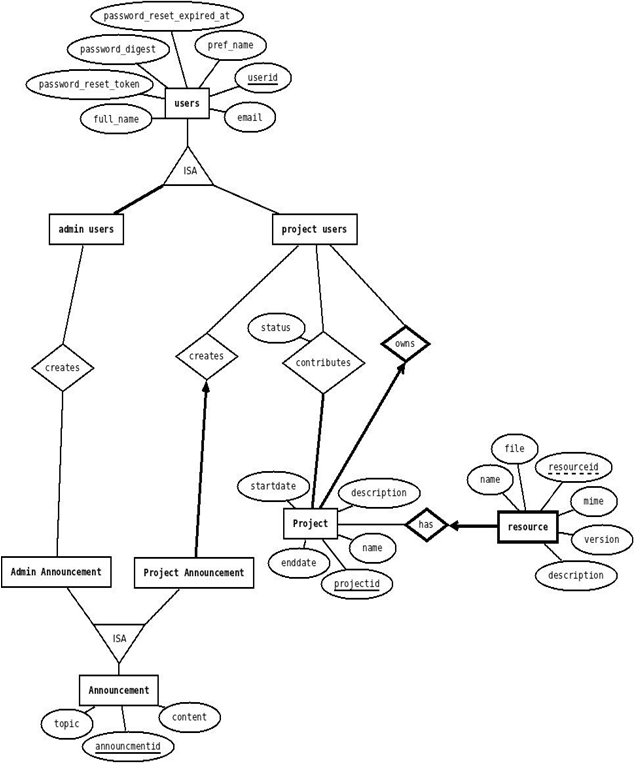


Figure: Entity-relationship diagram (see next section for detailed description)

**10.1 Primary tables**

We have used Ruby on Rails for the development with back end Sqllite3 database. This database has a convention on building the database. Particularly, each table has id, created\_at and updated\_at as default attributes to the tables. Hence we have used the convention when stating the tables below.

1. Users(userid, full\_name, pref\_name, email, admin, password\_digest, password\_reset\_token, password\_reset\_expired\_at)

Represents: entity set Users of the projects

Primary key: (userid)

Constraints: Users id should be unique, admin is true for administrators

1. Project(projectid , started\_at, ended\_at, name, description, ownerid)

Represents: entity set Project and relationship set owns

Primary key: (projectid)

Foreign key: (ownerid) references Project owner

Constraints: Project can be owned by a single user, but user can own multiple projects (project id is unique). Ownerid is not null.

1. Contributions (userid, projectid, status)

Represents: entity sets Users ,Project and relationship set Contributes

Primary key: (userid, projectid)

Foreign key: (userid) references Users, (projectid) references Project

Constraints: Project can have many contributors and contributor can work in many project. But user must request the project owner to contribute. By default user status 0. If user request owner for contribution, status 1 and when owner accept user to contribute status 2. Projectid is not null.

1. Resource(resourceid, projectid, name, version, description, file, mime)

Represents: entity sets Resource and Users and relationship set has

Primary key: (projectid, resourceid)

Foreign key: (projectid) references Project

Constraints: Project can own many resources. Projectid and resourceid is unique. Projectid is not null

1. Announcement(announcementid, userid, projectid, topic, content, kind)

Represents: entity sets Announcement

Primary key: (announcementid)

Foreign key: (userid) references Users, (projectid) references Project

Constraints: Announcement can be kind of Administrator or and project manager. Administrator can post announcements to all the users, but Project manager can post announcements to his contributors. Announcementid, projectid and userid is unique. Userid and projectid are not null, but projectid is 0 for admin users.

**10.2Tales and there dependency**

***Base Table***

* Users(userid, full\_name, presf\_name, email, admin, password\_digest, password\_reset\_token, password\_reset\_expired\_at)

***Dependent tables***

Hierarchical order of table, need to be entered

User

Announcement table

Project

Contributions

Resource table

**10.3 Functional Dependencies**

The following is a list of FD's for each table:

1. Users(userid, full\_name, pref\_name, email, admin, password\_digest, password\_reset\_token, password\_reset\_expired\_at)

* useridà full\_name, pref\_name, email, admin, password\_digest, password\_reset\_token, password\_reset\_expired\_at

1. Contributions (userid, projectid, status)

* userid, projectidà status
* projectid à userid
* userid à projectid
* userid à status

1. Project(projectid , started\_at, ended\_at, name, description, ownerid)

* projectid à started\_at, ended\_at, name, description
* projectid à ownerid

1. Resource(resourceid, projectid, name, version, description, file, mime)

* resourceid, projectid à name, version, description, file, mime
* resourceid à name, version, description, file, mime
* projectid à resourceid

5. Announcement(announcementid, userid, projectid, topic, content, kind)

* announcementid à topic, content, kind, userid, projectid
* announcementid, userid, projectid à topic, content, kind
* announcementid, userid, projectid à kind
* kind à topic, content, projectid

**10.4 Normalisation**

All the tables are in BCNF.

**11.0 Mobile platform specifications**

1. The mobile app is built over *PhoneGap* framework.
2. Supported features:
   1. Users can login to their account
   2. View their projects and Select a project
   3. Take a picture
   4. Choose a picture and upload it to their project
3. The app is tested in following configurations
   1. Android version - 2.3.4
   2. Device Model - HTC-Z710a

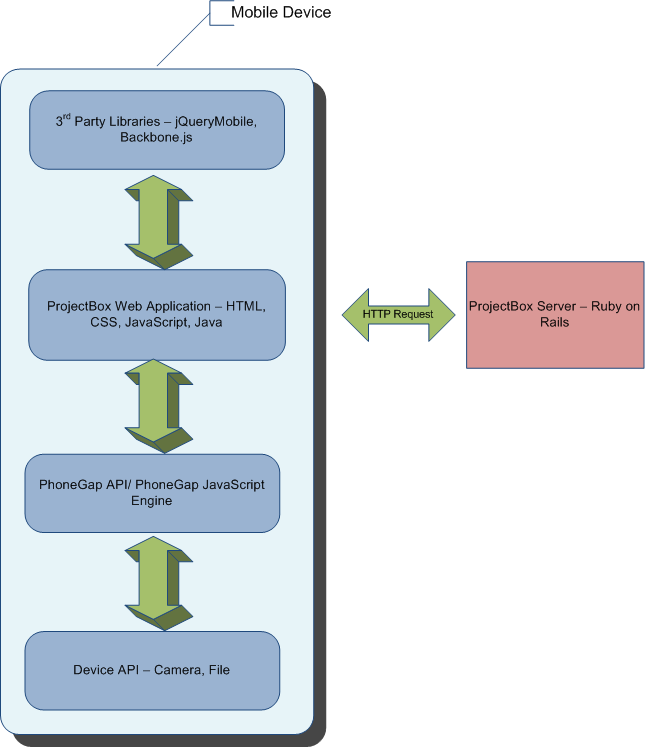


Figure: Interaction diagram of various layers in the mobile platform with rails server

**12.0 Authorisation and Authentication**

**12.1 Authentication System**

ProjectBox's authentication system is based on has\_secure\_password (ActiveModel::SecurePassword) of Rails 3.1. It offers following authentication functionality:

1. Makes attr\_accessor for password field
2. Provides confirmation validator of password
3. It hashes the password and stores it in the password\_digest field

**12.2 Encryption**

The system uses bcrypt-ruby hashing algorithm and it is a irreversible hashing mechanism.

**12.3 Authentication Process**

While authenticating the user, we use authenticate() function which takes unencrypted password. The password is hashed in runtime and compared with the password\_digest. If matches, the user is allowed else rejected.

**13.0 Overview of Tests**

Below is a list of Rspec tests (rspec spec --format documentation > ~/tests.txt):

AnnouncementsController

index

renders index template (FAILED - 1)

loads all the announcements (FAILED - 2)

show

renders show template (FAILED - 3)

finds and assigns the announcement (FAILED - 4)

new

creates a new announcement and assigns is (FAILED - 5)

new action should render new template (FAILED - 6)

create

renders new template when model is invalid (FAILED - 7)

redirect when model is valid (FAILED - 8)

edit

should render edit template (FAILED - 9)

update

should render edit template when model is invalid (FAILED - 10)

should redirect when model is valid (FAILED - 11)

destroy

destroys the model (FAILED - 12)

redirects to index action (FAILED - 13)

ProjectsController

index

renders index template

show

should render show template (FAILED - 14)

finds and assigns the project (FAILED - 15)

assigns contributors to project (FAILED - 16)

assigns resources to project (FAILED - 17)

new

creates a new project and assigns is (FAILED - 18)

new action should render new template (FAILED - 19)

create

should render new template when model is invalid (FAILED - 20)

should redirect when model is valid (FAILED - 21)

edit

should render edit template (FAILED - 22)

update

update action should render edit template when model is invalid (FAILED - 23)

update action should redirect when model is valid (FAILED - 24)

destroy

destroys the model (FAILED - 25)

redirects to the index action (FAILED - 26)

ResourcesController

index

renders index template (FAILED - 27)

loads all the resources (FAILED - 28)

show

renders show template (FAILED - 29)

finds and assigns the resource (FAILED - 30)

new

creates a new resource and assigns is (FAILED - 31)

new action should render new template (FAILED - 32)

create

should render new template when model is invalid (FAILED - 33)

should redirect when model is valid (FAILED - 34)

edit

should render edit template (FAILED - 35)

update

should render edit template when model is invalid (FAILED - 36)

should redirect when model is valid (FAILED - 37)

destroy

destroy the model (FAILED - 38)

redirects to the index action (FAILED - 39)

SessionsController

new

renders the new template if not logged in

redirects user if already logged in (FAILED - 40)

create

finds their user by email (FAILED - 41)

authenticates the user (FAILED - 42)

successful authentication

sets the users id in the session (FAILED - 43)

redirects to the users home page (FAILED - 44)

unsuccessful authentication

does not set the session id (FAILED - 45)

renders the new template (FAILED - 46)

destroy

it destroys the session token

redirects to the root url

UsersController

index

renders index template

loads all the users (FAILED - 47)

show

renders show template (FAILED - 48)

finds and assigns the user (FAILED - 49)

assigns the users managed projects (FAILED - 50)

assigns the users contributing projects (FAILED - 51)

new

creates a new user and assigns is (FAILED - 52)

new action should render new template (FAILED - 53)

create

create action should render new template when model is invalid (FAILED - 54)

create action should redirect when model is valid (FAILED - 55)

create action should set the sessions when model is valid (FAILED - 56)

edit

renders edit template (FAILED - 57)

update

update action should render edit template when model is invalid (FAILED - 58)

update action should redirect when model is valid (FAILED - 59)

destroy

destroys the model (FAILED - 60)

redirects to the index action (FAILED - 61)

MyResourcesHelper

add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/helpers/my\_resources\_helper\_spec.rb (PENDING: No reason given)

PasswordResetsHelper

add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/helpers/password\_resets\_helper\_spec.rb (PENDING: No reason given)

UserMailer

add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/mailers/user\_mailer\_spec.rb (PENDING: No reason given)

Ability

not\_admin

any user

can edit themselves (FAILED - 62)

can not edit another user (FAILED - 63)

can not view other user's associated projects (FAILED - 64)

manager

can edit owned project (FAILED - 65)

can view their associated projects (FAILED - 66)

cannot manage a random resource (FAILED - 67)

can delete other user's resource (FAILED - 68)

can replace other user's owned resource (FAILED - 69)

can modify other user's owned resource (FAILED - 70)

can download resource (FAILED - 71)

can upload resource (FAILED - 72)

can delete owned resource (FAILED - 73)

can replace owned resource (FAILED - 74)

can modify owned resource (FAILED - 75)

can remove a member from project (FAILED - 76)

contributor

starts out with contributor (FAILED - 77)

can not edit project (FAILED - 78)

can not delete other user's resource (FAILED - 79)

can not replace other user's owned resource (FAILED - 80)

can not modify other user's owned resource (FAILED - 81)

can download resource (FAILED - 82)

can upload resource (FAILED - 83)

can delete owned resource (FAILED - 84)

can replace owned resource (FAILED - 85)

can modify owned resource (FAILED - 86)

AdminUser

add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/models/admin\_user\_spec.rb (PENDING: No reason given)

Announcement

validations

starts out valid (FAILED - 87)

must have a topic (FAILED - 88)

must have well formatted topic (FAILED - 89)

must have a topic with minimum four characters (FAILED - 90)

must have content (FAILED - 91)

must have content with minimum 5 characters (FAILED - 92)

must have a kind (FAILED - 93)

Project

validations

starts out valid on creation (FAILED - 94)

must have a started date (FAILED - 95)

must have a ending date (FAILED - 96)

must have a name (FAILED - 97)

must have a description (FAILED - 98)

must have a owner (FAILED - 99)

must have well formatted name (FAILED - 100)

must have name that is at least 3 characters (FAILED - 101)

must have a unique name (FAILED - 102)

Resource

starts out valid on creation (FAILED - 103)

must have an owner (FAILED - 104)

must have a project\_id (FAILED - 105)

must have a name (FAILED - 106)

must have an alphanumeric name with no spaces (FAILED - 107)

must have a name with at least one letter or number (FAILED - 108)

must have a version number (FAILED - 109)

must have a description (FAILED - 110)

description must be at least 4 characters, and no larger than 256 (FAILED - 111)

must have a location (FAILED - 112)

must have a mime type (FAILED - 113)

must have a creation date (FAILED - 114)

must have a modified date (FAILED - 115)

must have a path (FAILED - 116)

name must be unique to a project (FAILED - 117)

User

validations

it starts out valid (FAILED - 118)

must have a email (FAILED - 119)

must have a full name (FAILED - 120)

must have well formatted name (FAILED - 121)

must have name that is at least 3 characters (FAILED - 122)

must have a well formatted email (FAILED - 123)

must have a unique email (FAILED - 124)

should start out as not an admin by default (FAILED - 125)

should not allow admin flag to be changed by mass assignment (FAILED - 126)

must have password that is at least 6 characters (FAILED - 127)

must confirm its password exactly (FAILED - 128)

authentication system

should return a user if password is correct (FAILED - 129)

should return FALSE if password is incorrect (FAILED - 130)

should create a password digest when saved (FAILED - 131)

admin use cases

managing projects

can create empty project (PENDING: Not yet implemented)

can delete any project (PENDING: Not yet implemented)

can manage contributors for a project (PENDING: Not yet implemented)

managing announcements

can create an announcement (PENDING: Not yet implemented)

can delete any announcement (PENDING: Not yet implemented)

can hide or show announcements on the front page (PENDING: Not yet implemented)

managing users

can send messages to users (PENDING: Not yet implemented)

can block users (PENDING: Not yet implemented)

can make other users into admin users (PENDING: Not yet implemented)

can edit users attributes (PENDING: Not yet implemented)

can manage user permissions (PENDING: Not yet implemented)

can remove user (PENDING: Not yet implemented)

contributor use cases

viewing projects

should list all the projects they are contributing to (PENDING: Not yet implemented)

should list the project names (PENDING: Not yet implemented)

should have links to the project pages (PENDING: Not yet implemented)

should list the members of each project (PENDING: Not yet implemented)

searching projects

should let them search by project name (PENDING: Not yet implemented)

should let them search by member name (PENDING: Not yet implemented)

should notify them when there are no search results (PENDING: Not yet implemented)

should have links to projects in results (PENDING: Not yet implemented)

downloading resource

should have a link to download from the project page (PENDING: Not yet implemented)

should display an error message if file not found (PENDING: Not yet implemented)

adding a resource

should show upload link on project page (PENDING: Not yet implemented)

should show download link after finishing upload (PENDING: Not yet implemented)

should show remove link after finishing upload (PENDING: Not yet implemented)

should show replace link after finishing upload (PENDING: Not yet implemented)

should show as belonging to the user that uploaded the resource (PENDING: Not yet implemented)

removing a resource

should be able to remove owned resource (PENDING: Not yet implemented)

should not be able to remove resources that aren't owned (PENDING: Not yet implemented)

should remove any other functionality from resource (PENDING: Not yet implemented)

replacing a resource

should be able replace owned resource (PENDING: Not yet implemented)

should not be able to replace resources that aren't own (PENDING: Not yet implemented)

creating a project

should provide creation link on their dashboard (PENDING: Not yet implemented)

should make this contributor a manager for the project (PENDING: Not yet implemented)

should create a new page for the project (PENDING: Not yet implemented)

should allow this manager to invite contributors (PENDING: Not yet implemented)

the log in process

has a link to the sign up page (FAILED - 132)

redirects a user to thier dashboard after login (FAILED - 133)

informs the user if thier log in credentials are incorrect (FAILED - 134)

manager use cases

managing contributors

can invite contributors to project (PENDING: Not yet implemented)

can kick-off contributors from project (PENDING: Not yet implemented)

can accept membership requests from contributors (PENDING: Not yet implemented)

can send notification to individual contributor (PENDING: Not yet implemented)

can view contributor profiles (PENDING: Not yet implemented)

managing resources

can add new resources to project (PENDING: Not yet implemented)

can delete any resources from project (PENDING: Not yet implemented)

replace resources in project (PENDING: Not yet implemented)

edit lifetime for resource (PENDING: Not yet implemented)

managing project

set project lietime (PENDING: Not yet implemented)

can delete a project that manager owns (PENDING: Not yet implemented)

can not delete a project that it does not own (PENDING: Not yet implemented)

can view notifications from contributors (PENDING: Not yet implemented)

can send notifications to all contributors (PENDING: Not yet implemented)

the resource handler

should allow a contributor and project owner to create a new resource (PENDING: Not yet implemented)

should allow a contributor to modify their resources only (PENDING: Not yet implemented)

should allow a project owner to modify any resource in the project (PENDING: Not yet implemented)

should allow contributor and project owner to download any resource in the project (PENDING: Not yet implemented)

visitor is not a member

should allow any visitor to download a resource (PENDING: Not yet implemented)

should not allow visitor to modify a resource (PENDING: Not yet implemented)

should not allow a visitor to create a new resource (PENDING: Not yet implemented)

visitor use cases

should present a choice to login or register (PENDING: Not yet implemented)

visitor is not a member

should present register page (PENDING: Not yet implemented)

should present the dashboard page after registration (PENDING: Not yet implemented)

visitor is a member

should present login page (PENDING: Not yet implemented)

should present the dashboard after loggin in (PENDING: Not yet implemented)

should list un-accepted invitations to projects from managers (PENDING: Not yet implemented)

should detect if member is a manager (PENDING: Not yet implemented)

should detect if member is a contributor (PENDING: Not yet implemented)

should have search projects option on dashboard (PENDING: Not yet implemented)

should allow membership request for selected project(s) (PENDING: Not yet implemented)

should allow visitor to create new project(s) (PENDING: Not yet implemented)

password\_resets/new.html.erb

add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/views/password\_resets/new.html.erb\_spec.rb (PENDING: No reason given)

Pending:

MyResourcesHelper add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/helpers/my\_resources\_helper\_spec.rb

# No reason given

# ./spec/helpers/my\_resources\_helper\_spec.rb:14

PasswordResetsHelper add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/helpers/password\_resets\_helper\_spec.rb

# No reason given

# ./spec/helpers/password\_resets\_helper\_spec.rb:14

UserMailer add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/mailers/user\_mailer\_spec.rb

# No reason given

# ./spec/mailers/user\_mailer\_spec.rb:4

AdminUser add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/models/admin\_user\_spec.rb

# No reason given

# ./spec/models/admin\_user\_spec.rb:4

admin use cases managing projects can create empty project

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:7

admin use cases managing projects can delete any project

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:9

admin use cases managing projects can manage contributors for a project

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:11

admin use cases managing announcements can create an announcement

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:18

admin use cases managing announcements can delete any announcement

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:20

admin use cases managing announcements can hide or show announcements on the front page

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:22

admin use cases managing users can send messages to users

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:29

admin use cases managing users can block users

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:31

admin use cases managing users can make other users into admin users

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:33

admin use cases managing users can edit users attributes

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:35

admin use cases managing users can manage user permissions

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:37

admin use cases managing users can remove user

# Not yet implemented

# ./spec/requests/administrator\_usecase\_spec.rb:39

contributor use cases viewing projects should list all the projects they are contributing to

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:7

contributor use cases viewing projects should list the project names

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:9

contributor use cases viewing projects should have links to the project pages

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:11

contributor use cases viewing projects should list the members of each project

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:13

contributor use cases searching projects should let them search by project name

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:19

contributor use cases searching projects should let them search by member name

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:21

contributor use cases searching projects should notify them when there are no search results

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:23

contributor use cases searching projects should have links to projects in results

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:25

contributor use cases downloading resource should have a link to download from the project page

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:31

contributor use cases downloading resource should display an error message if file not found

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:33

contributor use cases adding a resource should show upload link on project page

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:40

contributor use cases adding a resource should show download link after finishing upload

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:42

contributor use cases adding a resource should show remove link after finishing upload

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:44

contributor use cases adding a resource should show replace link after finishing upload

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:46

contributor use cases adding a resource should show as belonging to the user that uploaded the resource

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:48

contributor use cases removing a resource should be able to remove owned resource

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:55

contributor use cases removing a resource should not be able to remove resources that aren't owned

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:57

contributor use cases removing a resource should remove any other functionality from resource

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:59

contributor use cases replacing a resource should be able replace owned resource

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:65

contributor use cases replacing a resource should not be able to replace resources that aren't own

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:67

contributor use cases creating a project should provide creation link on their dashboard

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:74

contributor use cases creating a project should make this contributor a manager for the project

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:76

contributor use cases creating a project should create a new page for the project

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:78

contributor use cases creating a project should allow this manager to invite contributors

# Not yet implemented

# ./spec/requests/contributor\_usecase\_spec.rb:80

manager use cases managing contributors can invite contributors to project

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:7

manager use cases managing contributors can kick-off contributors from project

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:9

manager use cases managing contributors can accept membership requests from contributors

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:11

manager use cases managing contributors can send notification to individual contributor

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:13

manager use cases managing contributors can view contributor profiles

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:15

manager use cases managing resources can add new resources to project

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:22

manager use cases managing resources can delete any resources from project

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:24

manager use cases managing resources replace resources in project

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:26

manager use cases managing resources edit lifetime for resource

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:28

manager use cases managing project set project lietime

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:36

manager use cases managing project can delete a project that manager owns

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:38

manager use cases managing project can not delete a project that it does not own

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:40

manager use cases managing project can view notifications from contributors

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:42

manager use cases managing project can send notifications to all contributors

# Not yet implemented

# ./spec/requests/manager\_usecase\_spec.rb:44

the resource handler should allow a contributor and project owner to create a new resource

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:5

the resource handler should allow a contributor to modify their resources only

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:7

the resource handler should allow a project owner to modify any resource in the project

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:9

the resource handler should allow contributor and project owner to download any resource in the project

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:19

the resource handler visitor is not a member should allow any visitor to download a resource

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:12

the resource handler visitor is not a member should not allow visitor to modify a resource

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:14

the resource handler visitor is not a member should not allow a visitor to create a new resource

# Not yet implemented

# ./spec/requests/resource\_usecase\_spec.rb:16

visitor use cases should present a choice to login or register

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:5

visitor use cases visitor is not a member should present register page

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:9

visitor use cases visitor is not a member should present the dashboard page after registration

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:11

visitor use cases visitor is a member should present login page

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:18

visitor use cases visitor is a member should present the dashboard after loggin in

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:20

visitor use cases visitor is a member should list un-accepted invitations to projects from managers

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:22

visitor use cases visitor is a member should detect if member is a manager

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:24

visitor use cases visitor is a member should detect if member is a contributor

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:26

visitor use cases visitor is a member should have search projects option on dashboard

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:28

visitor use cases visitor is a member should allow membership request for selected project(s)

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:30

visitor use cases visitor is a member should allow visitor to create new project(s)

# Not yet implemented

# ./spec/requests/vistor\_usecase\_spec.rb:32

password\_resets/new.html.erb add some examples to (or delete) C:/Users/Parvez/cics530-430-cms/cms\_rails/spec/views/password\_resets/new.html.erb\_spec.rb

# No reason given

# ./spec/views/password\_resets/new.html.erb\_spec.rb:4

Failures:

1) AnnouncementsController index renders index template

Failure/Error: get "index"

ActionController::RoutingError:

No route matches {:controller=>"announcements"}

# ./spec/controllers/announcements\_controller\_spec.rb:13:in `block (3 levels) in <top (required)>'

2) AnnouncementsController index loads all the announcements

Failure/Error: get :index

ActionController::RoutingError:

No route matches {:controller=>"announcements"}

# ./spec/controllers/announcements\_controller\_spec.rb:18:in `block (3 levels) in <top (required)>'

3) AnnouncementsController show renders show template

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:25:in `block (3 levels) in <top (required)>'

4) AnnouncementsController show finds and assigns the announcement

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:25:in `block (3 levels) in <top (required)>'

5) AnnouncementsController new creates a new announcement and assigns is

Failure/Error: get :new

ActionController::RoutingError:

No route matches {:controller=>"announcements", :action=>"new"}

# ./spec/controllers/announcements\_controller\_spec.rb:42:in `block (3 levels) in <top (required)>'

6) AnnouncementsController new new action should render new template

Failure/Error: get :new

ActionController::RoutingError:

No route matches {:controller=>"announcements", :action=>"new"}

# ./spec/controllers/announcements\_controller\_spec.rb:46:in `block (3 levels) in <top (required)>'

7) AnnouncementsController create renders new template when model is invalid

Failure/Error: post :create

ActionController::RoutingError:

No route matches {:controller=>"announcements", :action=>"create"}

# ./spec/controllers/announcements\_controller\_spec.rb:54:in `block (3 levels) in <top (required)>'

8) AnnouncementsController create redirect when model is valid

Failure/Error: post :create

ActionController::RoutingError:

No route matches {:controller=>"announcements", :action=>"create"}

# ./spec/controllers/announcements\_controller\_spec.rb:60:in `block (3 levels) in <top (required)>'

9) AnnouncementsController edit should render edit template

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:67:in `block (3 levels) in <top (required)>'

10) AnnouncementsController update should render edit template when model is invalid

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:78:in `block (3 levels) in <top (required)>'

11) AnnouncementsController update should redirect when model is valid

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:78:in `block (3 levels) in <top (required)>'

12) AnnouncementsController destroy destroys the model

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:96:in `block (3 levels) in <top (required)>'

13) AnnouncementsController destroy redirects to index action

Failure/Error: @announcement = Factory(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/controllers/announcements\_controller\_spec.rb:96:in `block (3 levels) in <top (required)>'

14) ProjectsController show should render show template

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:22:in `block (3 levels) in <top (required)>'

15) ProjectsController show finds and assigns the project

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:22:in `block (3 levels) in <top (required)>'

16) ProjectsController show assigns contributors to project

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:22:in `block (3 levels) in <top (required)>'

17) ProjectsController show assigns resources to project

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:22:in `block (3 levels) in <top (required)>'

18) ProjectsController new creates a new project and assigns is

Failure/Error: get :new

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# ./app/controllers/projects\_controller.rb:23:in `new'

# ./app/controllers/projects\_controller.rb:23:in `new'

# ./spec/controllers/projects\_controller\_spec.rb:54:in `block (3 levels) in <top (required)>'

19) ProjectsController new new action should render new template

Failure/Error: get :new

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# ./app/controllers/projects\_controller.rb:23:in `new'

# ./app/controllers/projects\_controller.rb:23:in `new'

# ./spec/controllers/projects\_controller\_spec.rb:59:in `block (3 levels) in <top (required)>'

20) ProjectsController create should render new template when model is invalid

Failure/Error: post :create

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# ./app/controllers/projects\_controller.rb:27:in `new'

# ./app/controllers/projects\_controller.rb:27:in `create'

# ./spec/controllers/projects\_controller\_spec.rb:67:in `block (3 levels) in <top (required)>'

21) ProjectsController create should redirect when model is valid

Failure/Error: post :create

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# ./app/controllers/projects\_controller.rb:27:in `new'

# ./app/controllers/projects\_controller.rb:27:in `create'

# ./spec/controllers/projects\_controller\_spec.rb:73:in `block (3 levels) in <top (required)>'

22) ProjectsController edit should render edit template

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:81:in `block (3 levels) in <top (required)>'

23) ProjectsController update update action should render edit template when model is invalid

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:94:in `block (3 levels) in <top (required)>'

24) ProjectsController update update action should redirect when model is valid

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:94:in `block (3 levels) in <top (required)>'

25) ProjectsController destroy destroys the model

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:113:in `block (3 levels) in <top (required)>'

26) ProjectsController destroy redirects to the index action

Failure/Error: @project = Factory(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/controllers/projects\_controller\_spec.rb:113:in `block (3 levels) in <top (required)>'

27) ResourcesController index renders index template

Failure/Error: get :index

ActionController::RoutingError:

No route matches {:controller=>"resources"}

# ./spec/controllers/resources\_controller\_spec.rb:13:in `block (3 levels) in <top (required)>'

28) ResourcesController index loads all the resources

Failure/Error: get :index

ActionController::RoutingError:

No route matches {:controller=>"resources"}

# ./spec/controllers/resources\_controller\_spec.rb:19:in `block (3 levels) in <top (required)>'

29) ResourcesController show renders show template

Failure/Error: @resource = Factory(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:29:in `block (3 levels) in <top (required)>'

30) ResourcesController show finds and assigns the resource

Failure/Error: @resource = Factory(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:29:in `block (3 levels) in <top (required)>'

31) ResourcesController new creates a new resource and assigns is

Failure/Error: get :new

ActionController::RoutingError:

No route matches {:controller=>"resources", :action=>"new"}

# ./spec/controllers/resources\_controller\_spec.rb:51:in `block (3 levels) in <top (required)>'

32) ResourcesController new new action should render new template

Failure/Error: get :new

ActionController::RoutingError:

No route matches {:controller=>"resources", :action=>"new"}

# ./spec/controllers/resources\_controller\_spec.rb:56:in `block (3 levels) in <top (required)>'

33) ResourcesController create should render new template when model is invalid

Failure/Error: post :create

ActionController::RoutingError:

No route matches {:controller=>"resources", :action=>"create"}

# ./spec/controllers/resources\_controller\_spec.rb:65:in `block (3 levels) in <top (required)>'

34) ResourcesController create should redirect when model is valid

Failure/Error: post :create

ActionController::RoutingError:

No route matches {:controller=>"resources", :action=>"create"}

# ./spec/controllers/resources\_controller\_spec.rb:71:in `block (3 levels) in <top (required)>'

35) ResourcesController edit should render edit template

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:80:in `block (3 levels) in <top (required)>'

36) ResourcesController update should render edit template when model is invalid

Failure/Error: @resource = Factory(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:93:in `block (3 levels) in <top (required)>'

37) ResourcesController update should redirect when model is valid

Failure/Error: @resource = Factory(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:93:in `block (3 levels) in <top (required)>'

38) ResourcesController destroy destroy the model

Failure/Error: @resource = Factory(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:113:in `block (3 levels) in <top (required)>'

39) ResourcesController destroy redirects to the index action

Failure/Error: @resource = Factory(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/controllers/resources\_controller\_spec.rb:113:in `block (3 levels) in <top (required)>'

40) SessionsController new redirects user if already logged in

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:13:in `block (3 levels) in <top (required)>'

41) SessionsController create finds their user by email

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:24:in `block (3 levels) in <top (required)>'

42) SessionsController create authenticates the user

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:24:in `block (3 levels) in <top (required)>'

43) SessionsController create successful authentication sets the users id in the session

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:24:in `block (3 levels) in <top (required)>'

44) SessionsController create successful authentication redirects to the users home page

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:24:in `block (3 levels) in <top (required)>'

45) SessionsController create unsuccessful authentication does not set the session id

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:24:in `block (3 levels) in <top (required)>'

46) SessionsController create unsuccessful authentication renders the new template

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/sessions\_controller\_spec.rb:24:in `block (3 levels) in <top (required)>'

47) UsersController index loads all the users

Failure/Error: User.expects(:all).returns([])

Mocha::ExpectationError:

not all expectations were satisfied

unsatisfied expectations:

- expected exactly once, not yet invoked: User(Table doesn't exist).all(any\_parameters)

satisfied expectations:

- allowed any number of times, invoked once: #<UsersController:0x5228f10>.is\_logged\_in(any\_parameters)

# ./spec/controllers/users\_controller\_spec.rb:18:in `block (3 levels) in <top (required)>'

48) UsersController show renders show template

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:28:in `block (3 levels) in <top (required)>'

49) UsersController show finds and assigns the user

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:28:in `block (3 levels) in <top (required)>'

50) UsersController show assigns the users managed projects

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:28:in `block (3 levels) in <top (required)>'

51) UsersController show assigns the users contributing projects

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:28:in `block (3 levels) in <top (required)>'

52) UsersController new creates a new user and assigns is

Failure/Error: get :new

ActiveRecord::StatementInvalid:

Could not find table 'users'

# ./app/controllers/users\_controller.rb:33:in `new'

# ./app/controllers/users\_controller.rb:33:in `new'

# ./spec/controllers/users\_controller\_spec.rb:62:in `block (3 levels) in <top (required)>'

53) UsersController new new action should render new template

Failure/Error: get :new

ActiveRecord::StatementInvalid:

Could not find table 'users'

# ./app/controllers/users\_controller.rb:33:in `new'

# ./app/controllers/users\_controller.rb:33:in `new'

# ./spec/controllers/users\_controller\_spec.rb:67:in `block (3 levels) in <top (required)>'

54) UsersController create create action should render new template when model is invalid

Failure/Error: post :create

ActiveRecord::StatementInvalid:

Could not find table 'users'

# ./app/controllers/users\_controller.rb:39:in `new'

# ./app/controllers/users\_controller.rb:39:in `create'

# ./spec/controllers/users\_controller\_spec.rb:76:in `block (3 levels) in <top (required)>'

55) UsersController create create action should redirect when model is valid

Failure/Error: post :create

ActiveRecord::StatementInvalid:

Could not find table 'users'

# ./app/controllers/users\_controller.rb:39:in `new'

# ./app/controllers/users\_controller.rb:39:in `create'

# ./spec/controllers/users\_controller\_spec.rb:82:in `block (3 levels) in <top (required)>'

56) UsersController create create action should set the sessions when model is valid

Failure/Error: post :create

ActiveRecord::StatementInvalid:

Could not find table 'users'

# ./app/controllers/users\_controller.rb:39:in `new'

# ./app/controllers/users\_controller.rb:39:in `create'

# ./spec/controllers/users\_controller\_spec.rb:88:in `block (3 levels) in <top (required)>'

57) UsersController edit renders edit template

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:96:in `block (3 levels) in <top (required)>'

58) UsersController update update action should render edit template when model is invalid

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:109:in `block (3 levels) in <top (required)>'

59) UsersController update update action should redirect when model is valid

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:109:in `block (3 levels) in <top (required)>'

60) UsersController destroy destroys the model

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:129:in `block (3 levels) in <top (required)>'

61) UsersController destroy redirects to the index action

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/controllers/users\_controller\_spec.rb:129:in `block (3 levels) in <top (required)>'

62) Ability not\_admin any user can edit themselves

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

63) Ability not\_admin any user can not edit another user

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

64) Ability not\_admin any user can not view other user's associated projects

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

65) Ability not\_admin manager can edit owned project

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

66) Ability not\_admin manager can view their associated projects

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

67) Ability not\_admin manager cannot manage a random resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

68) Ability not\_admin manager can delete other user's resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

69) Ability not\_admin manager can replace other user's owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

70) Ability not\_admin manager can modify other user's owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

71) Ability not\_admin manager can download resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

72) Ability not\_admin manager can upload resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

73) Ability not\_admin manager can delete owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

74) Ability not\_admin manager can replace owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

75) Ability not\_admin manager can modify owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

76) Ability not\_admin manager can remove a member from project

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

77) Ability not\_admin contributor starts out with contributor

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

78) Ability not\_admin contributor can not edit project

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

79) Ability not\_admin contributor can not delete other user's resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

80) Ability not\_admin contributor can not replace other user's owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

81) Ability not\_admin contributor can not modify other user's owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

82) Ability not\_admin contributor can download resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

83) Ability not\_admin contributor can upload resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

84) Ability not\_admin contributor can delete owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

85) Ability not\_admin contributor can replace owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

86) Ability not\_admin contributor can modify owned resource

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/ability\_spec.rb:16:in `block (3 levels) in <top (required)>'

87) Announcement validations starts out valid

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

88) Announcement validations must have a topic

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

89) Announcement validations must have well formatted topic

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

90) Announcement validations must have a topic with minimum four characters

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

91) Announcement validations must have content

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

92) Announcement validations must have content with minimum 5 characters

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

93) Announcement validations must have a kind

Failure/Error: @announcement = Factory.build(:announcement)

ActiveRecord::StatementInvalid:

Could not find table 'announcements'

# C:in `object'

# ./spec/models/announcement\_spec.rb:7:in `block (3 levels) in <top (required)>'

94) Project validations starts out valid on creation

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

95) Project validations must have a started date

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

96) Project validations must have a ending date

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

97) Project validations must have a name

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

98) Project validations must have a description

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

99) Project validations must have a owner

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

100) Project validations must have well formatted name

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

101) Project validations must have name that is at least 3 characters

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

102) Project validations must have a unique name

Failure/Error: @project = Factory.build(:project)

ActiveRecord::StatementInvalid:

Could not find table 'projects'

# C:in `object'

# ./spec/models/project\_spec.rb:8:in `block (3 levels) in <top (required)>'

103) Resource starts out valid on creation

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

104) Resource must have an owner

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

105) Resource must have a project\_id

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

106) Resource must have a name

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

107) Resource must have an alphanumeric name with no spaces

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

108) Resource must have a name with at least one letter or number

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

109) Resource must have a version number

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

110) Resource must have a description

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

111) Resource description must be at least 4 characters, and no larger than 256

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

112) Resource must have a location

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

113) Resource must have a mime type

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

114) Resource must have a creation date

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

115) Resource must have a modified date

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

116) Resource must have a path

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

117) Resource name must be unique to a project

Failure/Error: @resource = Factory.build(:resource)

ActiveRecord::StatementInvalid:

Could not find table 'resources'

# C:in `object'

# ./spec/models/resource\_spec.rb:5:in `block (2 levels) in <top (required)>'

118) User validations it starts out valid

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

119) User validations must have a email

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

120) User validations must have a full name

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

121) User validations must have well formatted name

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

122) User validations must have name that is at least 3 characters

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

123) User validations must have a well formatted email

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

124) User validations must have a unique email

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

125) User validations should start out as not an admin by default

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

126) User validations should not allow admin flag to be changed by mass assignment

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

127) User validations must have password that is at least 6 characters

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

128) User validations must confirm its password exactly

Failure/Error: @user = Factory.build(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:8:in `block (3 levels) in <top (required)>'

129) User authentication system should return a user if password is correct

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:85:in `block (3 levels) in <top (required)>'

130) User authentication system should return FALSE if password is incorrect

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:85:in `block (3 levels) in <top (required)>'

131) User authentication system should create a password digest when saved

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/models/user\_spec.rb:85:in `block (3 levels) in <top (required)>'

132) the log in process has a link to the sign up page

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/requests/login\_spec.rb:6:in `block (2 levels) in <top (required)>'

133) the log in process redirects a user to thier dashboard after login

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/requests/login\_spec.rb:6:in `block (2 levels) in <top (required)>'

134) the log in process informs the user if thier log in credentials are incorrect

Failure/Error: @user = Factory(:user)

ActiveRecord::StatementInvalid:

Could not find table 'users'

# C:in `object'

# ./spec/requests/login\_spec.rb:6:in `block (2 levels) in <top (required)>'

Finished in 5.13 seconds

212 examples, 134 failures, 73 pending

Failed examples:

rspec ./spec/controllers/announcements\_controller\_spec.rb:12 # AnnouncementsController index renders index template

rspec ./spec/controllers/announcements\_controller\_spec.rb:16 # AnnouncementsController index loads all the announcements

rspec ./spec/controllers/announcements\_controller\_spec.rb:30 # AnnouncementsController show renders show template

rspec ./spec/controllers/announcements\_controller\_spec.rb:34 # AnnouncementsController show finds and assigns the announcement

rspec ./spec/controllers/announcements\_controller\_spec.rb:41 # AnnouncementsController new creates a new announcement and assigns is

rspec ./spec/controllers/announcements\_controller\_spec.rb:45 # AnnouncementsController new new action should render new template

rspec ./spec/controllers/announcements\_controller\_spec.rb:52 # AnnouncementsController create renders new template when model is invalid

rspec ./spec/controllers/announcements\_controller\_spec.rb:58 # AnnouncementsController create redirect when model is valid

rspec ./spec/controllers/announcements\_controller\_spec.rb:70 # AnnouncementsController edit should render edit template

rspec ./spec/controllers/announcements\_controller\_spec.rb:80 # AnnouncementsController update should render edit template when model is invalid

rspec ./spec/controllers/announcements\_controller\_spec.rb:86 # AnnouncementsController update should redirect when model is valid

rspec ./spec/controllers/announcements\_controller\_spec.rb:99 # AnnouncementsController destroy destroys the model

rspec ./spec/controllers/announcements\_controller\_spec.rb:104 # AnnouncementsController destroy redirects to index action

rspec ./spec/controllers/projects\_controller\_spec.rb:28 # ProjectsController show should render show template

rspec ./spec/controllers/projects\_controller\_spec.rb:33 # ProjectsController show finds and assigns the project

rspec ./spec/controllers/projects\_controller\_spec.rb:38 # ProjectsController show assigns contributors to project

rspec ./spec/controllers/projects\_controller\_spec.rb:44 # ProjectsController show assigns resources to project

rspec ./spec/controllers/projects\_controller\_spec.rb:53 # ProjectsController new creates a new project and assigns is

rspec ./spec/controllers/projects\_controller\_spec.rb:58 # ProjectsController new new action should render new template

rspec ./spec/controllers/projects\_controller\_spec.rb:65 # ProjectsController create should render new template when model is invalid

rspec ./spec/controllers/projects\_controller\_spec.rb:71 # ProjectsController create should redirect when model is valid

rspec ./spec/controllers/projects\_controller\_spec.rb:85 # ProjectsController edit should render edit template

rspec ./spec/controllers/projects\_controller\_spec.rb:97 # ProjectsController update update action should render edit template when model is invalid

rspec ./spec/controllers/projects\_controller\_spec.rb:103 # ProjectsController update update action should redirect when model is valid

rspec ./spec/controllers/projects\_controller\_spec.rb:116 # ProjectsController destroy destroys the model

rspec ./spec/controllers/projects\_controller\_spec.rb:122 # ProjectsController destroy redirects to the index action

rspec ./spec/controllers/resources\_controller\_spec.rb:12 # ResourcesController index renders index template

rspec ./spec/controllers/resources\_controller\_spec.rb:17 # ResourcesController index loads all the resources

rspec ./spec/controllers/resources\_controller\_spec.rb:35 # ResourcesController show renders show template

rspec ./spec/controllers/resources\_controller\_spec.rb:40 # ResourcesController show finds and assigns the resource

rspec ./spec/controllers/resources\_controller\_spec.rb:50 # ResourcesController new creates a new resource and assigns is

rspec ./spec/controllers/resources\_controller\_spec.rb:55 # ResourcesController new new action should render new template

rspec ./spec/controllers/resources\_controller\_spec.rb:63 # ResourcesController create should render new template when model is invalid

rspec ./spec/controllers/resources\_controller\_spec.rb:69 # ResourcesController create should redirect when model is valid

rspec ./spec/controllers/resources\_controller\_spec.rb:84 # ResourcesController edit should render edit template

rspec ./spec/controllers/resources\_controller\_spec.rb:96 # ResourcesController update should render edit template when model is invalid

rspec ./spec/controllers/resources\_controller\_spec.rb:102 # ResourcesController update should redirect when model is valid

rspec ./spec/controllers/resources\_controller\_spec.rb:116 # ResourcesController destroy destroy the model

rspec ./spec/controllers/resources\_controller\_spec.rb:123 # ResourcesController destroy redirects to the index action

rspec ./spec/controllers/sessions\_controller\_spec.rb:12 # SessionsController new redirects user if already logged in

rspec ./spec/controllers/sessions\_controller\_spec.rb:28 # SessionsController create finds their user by email

rspec ./spec/controllers/sessions\_controller\_spec.rb:33 # SessionsController create authenticates the user

rspec ./spec/controllers/sessions\_controller\_spec.rb:46 # SessionsController create successful authentication sets the users id in the session

rspec ./spec/controllers/sessions\_controller\_spec.rb:51 # SessionsController create successful authentication redirects to the users home page

rspec ./spec/controllers/sessions\_controller\_spec.rb:65 # SessionsController create unsuccessful authentication does not set the session id

rspec ./spec/controllers/sessions\_controller\_spec.rb:70 # SessionsController create unsuccessful authentication renders the new template

rspec ./spec/controllers/users\_controller\_spec.rb:17 # UsersController index loads all the users

rspec ./spec/controllers/users\_controller\_spec.rb:35 # UsersController show renders show template

rspec ./spec/controllers/users\_controller\_spec.rb:40 # UsersController show finds and assigns the user

rspec ./spec/controllers/users\_controller\_spec.rb:45 # UsersController show assigns the users managed projects

rspec ./spec/controllers/users\_controller\_spec.rb:51 # UsersController show assigns the users contributing projects

rspec ./spec/controllers/users\_controller\_spec.rb:61 # UsersController new creates a new user and assigns is

rspec ./spec/controllers/users\_controller\_spec.rb:66 # UsersController new new action should render new template

rspec ./spec/controllers/users\_controller\_spec.rb:74 # UsersController create create action should render new template when model is invalid

rspec ./spec/controllers/users\_controller\_spec.rb:80 # UsersController create create action should redirect when model is valid

rspec ./spec/controllers/users\_controller\_spec.rb:86 # UsersController create create action should set the sessions when model is valid

rspec ./spec/controllers/users\_controller\_spec.rb:100 # UsersController edit renders edit template

rspec ./spec/controllers/users\_controller\_spec.rb:112 # UsersController update update action should render edit template when model is invalid

rspec ./spec/controllers/users\_controller\_spec.rb:118 # UsersController update update action should redirect when model is valid

rspec ./spec/controllers/users\_controller\_spec.rb:132 # UsersController destroy destroys the model

rspec ./spec/controllers/users\_controller\_spec.rb:138 # UsersController destroy redirects to the index action

rspec ./spec/models/ability\_spec.rb:25 # Ability not\_admin any user can edit themselves

rspec ./spec/models/ability\_spec.rb:29 # Ability not\_admin any user can not edit another user

rspec ./spec/models/ability\_spec.rb:33 # Ability not\_admin any user can not view other user's associated projects

rspec ./spec/models/ability\_spec.rb:54 # Ability not\_admin manager can edit owned project

rspec ./spec/models/ability\_spec.rb:58 # Ability not\_admin manager can view their associated projects

rspec ./spec/models/ability\_spec.rb:62 # Ability not\_admin manager cannot manage a random resource

rspec ./spec/models/ability\_spec.rb:68 # Ability not\_admin manager can delete other user's resource

rspec ./spec/models/ability\_spec.rb:72 # Ability not\_admin manager can replace other user's owned resource

rspec ./spec/models/ability\_spec.rb:76 # Ability not\_admin manager can modify other user's owned resource

rspec ./spec/models/ability\_spec.rb:80 # Ability not\_admin manager can download resource

rspec ./spec/models/ability\_spec.rb:84 # Ability not\_admin manager can upload resource

rspec ./spec/models/ability\_spec.rb:88 # Ability not\_admin manager can delete owned resource

rspec ./spec/models/ability\_spec.rb:93 # Ability not\_admin manager can replace owned resource

rspec ./spec/models/ability\_spec.rb:97 # Ability not\_admin manager can modify owned resource

rspec ./spec/models/ability\_spec.rb:101 # Ability not\_admin manager can remove a member from project

rspec ./spec/models/ability\_spec.rb:120 # Ability not\_admin contributor starts out with contributor

rspec ./spec/models/ability\_spec.rb:124 # Ability not\_admin contributor can not edit project

rspec ./spec/models/ability\_spec.rb:128 # Ability not\_admin contributor can not delete other user's resource

rspec ./spec/models/ability\_spec.rb:132 # Ability not\_admin contributor can not replace other user's owned resource

rspec ./spec/models/ability\_spec.rb:135 # Ability not\_admin contributor can not modify other user's owned resource

rspec ./spec/models/ability\_spec.rb:139 # Ability not\_admin contributor can download resource

rspec ./spec/models/ability\_spec.rb:143 # Ability not\_admin contributor can upload resource

rspec ./spec/models/ability\_spec.rb:147 # Ability not\_admin contributor can delete owned resource

rspec ./spec/models/ability\_spec.rb:151 # Ability not\_admin contributor can replace owned resource

rspec ./spec/models/ability\_spec.rb:155 # Ability not\_admin contributor can modify owned resource

rspec ./spec/models/announcement\_spec.rb:10 # Announcement validations starts out valid

rspec ./spec/models/announcement\_spec.rb:15 # Announcement validations must have a topic

rspec ./spec/models/announcement\_spec.rb:20 # Announcement validations must have well formatted topic

rspec ./spec/models/announcement\_spec.rb:25 # Announcement validations must have a topic with minimum four characters

rspec ./spec/models/announcement\_spec.rb:30 # Announcement validations must have content

rspec ./spec/models/announcement\_spec.rb:35 # Announcement validations must have content with minimum 5 characters

rspec ./spec/models/announcement\_spec.rb:40 # Announcement validations must have a kind

rspec ./spec/models/project\_spec.rb:11 # Project validations starts out valid on creation

rspec ./spec/models/project\_spec.rb:21 # Project validations must have a started date

rspec ./spec/models/project\_spec.rb:26 # Project validations must have a ending date

rspec ./spec/models/project\_spec.rb:31 # Project validations must have a name

rspec ./spec/models/project\_spec.rb:36 # Project validations must have a description

rspec ./spec/models/project\_spec.rb:41 # Project validations must have a owner

rspec ./spec/models/project\_spec.rb:46 # Project validations must have well formatted name

rspec ./spec/models/project\_spec.rb:51 # Project validations must have name that is at least 3 characters

rspec ./spec/models/project\_spec.rb:56 # Project validations must have a unique name

rspec ./spec/models/resource\_spec.rb:8 # Resource starts out valid on creation

rspec ./spec/models/resource\_spec.rb:13 # Resource must have an owner

rspec ./spec/models/resource\_spec.rb:18 # Resource must have a project\_id

rspec ./spec/models/resource\_spec.rb:23 # Resource must have a name

rspec ./spec/models/resource\_spec.rb:28 # Resource must have an alphanumeric name with no spaces

rspec ./spec/models/resource\_spec.rb:33 # Resource must have a name with at least one letter or number

rspec ./spec/models/resource\_spec.rb:38 # Resource must have a version number

rspec ./spec/models/resource\_spec.rb:43 # Resource must have a description

rspec ./spec/models/resource\_spec.rb:48 # Resource description must be at least 4 characters, and no larger than 256

rspec ./spec/models/resource\_spec.rb:53 # Resource must have a location

rspec ./spec/models/resource\_spec.rb:58 # Resource must have a mime type

rspec ./spec/models/resource\_spec.rb:63 # Resource must have a creation date

rspec ./spec/models/resource\_spec.rb:68 # Resource must have a modified date

rspec ./spec/models/resource\_spec.rb:78 # Resource must have a path

rspec ./spec/models/resource\_spec.rb:88 # Resource name must be unique to a project

rspec ./spec/models/user\_spec.rb:11 # User validations it starts out valid

rspec ./spec/models/user\_spec.rb:22 # User validations must have a email

rspec ./spec/models/user\_spec.rb:27 # User validations must have a full name

rspec ./spec/models/user\_spec.rb:32 # User validations must have well formatted name

rspec ./spec/models/user\_spec.rb:37 # User validations must have name that is at least 3 characters

rspec ./spec/models/user\_spec.rb:42 # User validations must have a well formatted email

rspec ./spec/models/user\_spec.rb:47 # User validations must have a unique email

rspec ./spec/models/user\_spec.rb:53 # User validations should start out as not an admin by default

rspec ./spec/models/user\_spec.rb:57 # User validations should not allow admin flag to be changed by mass assignment

rspec ./spec/models/user\_spec.rb:68 # User validations must have password that is at least 6 characters

rspec ./spec/models/user\_spec.rb:75 # User validations must confirm its password exactly

rspec ./spec/models/user\_spec.rb:89 # User authentication system should return a user if password is correct

rspec ./spec/models/user\_spec.rb:93 # User authentication system should return FALSE if password is incorrect

rspec ./spec/models/user\_spec.rb:97 # User authentication system should create a password digest when saved

rspec ./spec/requests/login\_spec.rb:11 # the log in process has a link to the sign up page

rspec ./spec/requests/login\_spec.rb:17 # the log in process redirects a user to thier dashboard after login

rspec ./spec/requests/login\_spec.rb:24 # the log in process informs the user if thier log in credentials are incorrect

**14.0 Known bugs (demo list – have to be redone)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index # | Bug type | Priority | Description | Date found |
| 1 | UI | Low | Chrome browser breaking the CSS | 22-04-2012 |
| 2 | Functional | Medium |  | 22-04-2012 |
| 3 | OS related | High |  | 22-04-2012 |
| 4 | DB | High |  | 22-04-2012 |
| 5 | Wrong test case | Low |  | 22-04-2012 |
| 6 | UI |  |  | 22-04-2012 |
| 7 | Nonfunctional | Medium |  | 22-04-2012 |

**15.0 Lessons learnt**

1. Rails is a fantastic platform for building applications quickly and efficiently
2. Rails in combination with Agile Scrum approach (which we roughly followed) to testing is a very good way to get applications up and running in a small time frame