# Access Review

Sponsor: Coupa Software
Point of Contact: Philip Cox
Advisor: Patrick Tague
Team Member: Kaiyu Liu, Rundong Liu, Haoran Liu

# Agenda

- Project goals
- Planned tasks to deliver
- Evaluation approaches
- Progress status
  - Technology applied
  - Techniques under construction and progress
  - System design and development progress
- Preliminary results
- Health of the project progress
- Plan for the rest of project

# Project overview

- Coupa software has many services for internal use. Different employees have different permissions for applications and private data.
- In this Access Review project, our team is required to come out of a solution to make access review process easy to manage and more secure.

http://accessreview.ca/

# Project Goal

- Build a web application to handle role based access review process.
- Visualize access control list to facilitate permission review and audit.
- Refine Database schema and other design for a better normalization.
- Generate pdf report after review process is completed.

# Planned Tasks to deliver

1. Parse csv files for different service with the users and their access permission into database
2. Provide authentication and session management for different roles.
3. Display diverse content based on user’s identity and their permission of backend data.
4. Provide web console for administrator to fully control roles and data of the system
5. Provide visual interaction interface for access review process
6. Provide a method to generate PDF version of the access review report.
7. Record update history for system data, append log file based on users’ action for future analysis.

# Evaluation Approaches

- Consumer-oriented
  - Discussed with Coupa software in their HQ in detail to know the requirement  of the company and best method to design the project
- Development Testing
  - Testing projects during development process
- Customer Review
  - Submit different versions of code for customer review and comments during development

# Progress Status

- Theory/technology applied
- Techniques under construction and progress
- System design and development progress

# Technology applied

1. Build role based access control list
2. Designed database schema based on customer requirement and data supplied (CSV files) provided by Coupa Software
3. Use Python/Django framework
4. Agile Development

# Techniques under construction and progress

Iteration 1

| Id | Description | Progress |
|---|---|---|
| 1.1 | Design database storage Schema | completed |
| 1.2 | Design system workflow | completed |
| 1.3 | Parse csv file, extract data and save into database | completed |
| 1.4 | Design basic wireframe for the web interface | completed |

# Techniques under construction and progress

## Iteration 2

| ID | Description | Progress |
|---|---|---|
| 2.1 | Separate different roles like admin, auditor and managers | Completed |
| 2.2 | Provide authentication mechanism to validate users | Completed |
| 2.3 | Build the interface for users to login the system | Completed |
| 2.4 | Build the interface to show the services in the dashboard for different users | Completed |
| 2.5 | Build interface to show the review pages for users | Completed |
| 2.6 | Test for the application in this stage | Completed |

# System Design

Roles outline

1. There are five kinds of entities Admin, Auditor, Manager, Users and Applications in this app
2. Admin is the one that manage the whole application. There is only one admin in the app
3. Managers are assigned for specific service, they can approve or deny the permissions for users
4. Auditor can only view the access permissions for all services

# System Design

Model:

- Admin -The one that controls the whole app
- User -The normal user in the app
- Auditor -The one that audit all the applications
- Manager -The one that reviews applications assigned to him
- Application -Services  involved in the company

# System Design

Relation

- App-User-Permission

  Access control list shows user permission

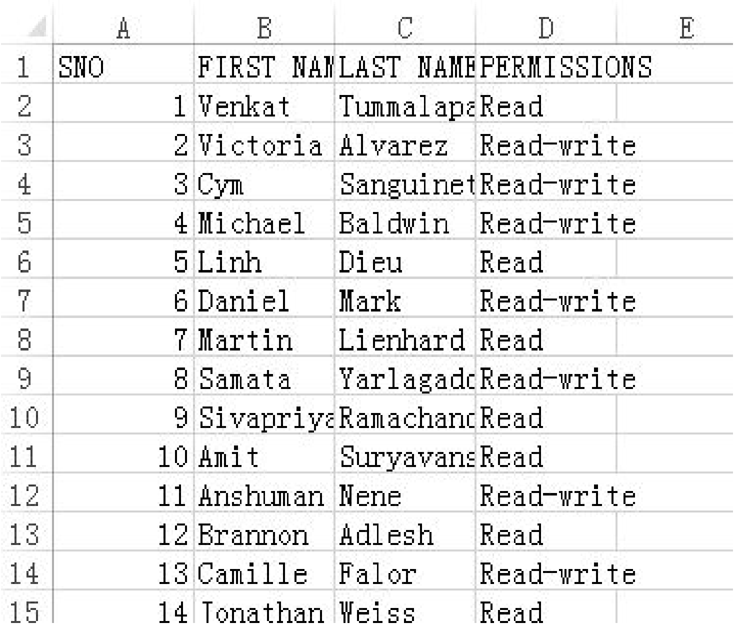
- App-Manager

  Show which manager is assigned to an app

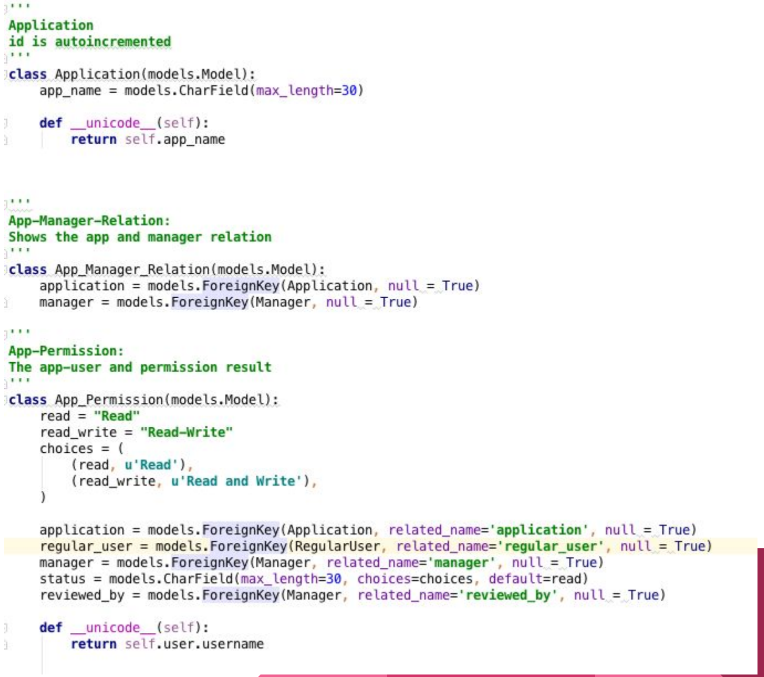
- App-Auditor

  Show which auditor is assigned to an app

# System Design



| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | SNO | FIRST NAM | LAST NAME | PERMISSIONS | |
| 2 | 1 | Venkat | Tummalapa | Read | |
| 3 | 2 | Victoria | Alvarez | Read-write | |
| 4 | 3 | Cym | Sanguinet | Read-write | |
| 5 | 4 | Michael | Baldwin | Read-write | |
| 6 | 5 | Linh | Dieu | Read | |
| 7 | 6 | Daniel | Mark | Read-write | |
| 8 | 7 | Martin | Lienhard | Read | |
| 9 | 8 | Samata | Yarlagadd | Read-write | |
| 10 | 9 | Sivapriya | Ramachand | Read | |
| 11 | 10 | Amit | Suryavans | Read | |
| 12 | 11 | Anshuman | Nene | Read-write | |
| 13 | 12 | Brannon | Adlesh | Read | |
| 14 | 13 | Camille | Falor | Read-write | |
| 15 | 14 | Jonathan | Weiss | Read | |



```
'''
Application
id is autoincremented
'''
class Application(models.Model):
    app_name = models.CharField(max_length=30)

    def __unicode__(self):
        return self.app_name


'''
App-Manager-Relation:
Shows the app and manager relation
'''
class App_Manager_Relation(models.Model):
    application = models.ForeignKey(Application, null = True)
    manager = models.ForeignKey(Manager, null = True)

'''
App-Permission:
The app-user and permission result
'''
class App_Permission(models.Model):
    read = "Read"
    read_write = "Read-Write"
    choices = (
        (read, u'Read'),
        (read_write, u'Read and Write'),
    )

    application = models.ForeignKey(Application, related_name='application', null = True)
    regular_user = models.ForeignKey(RegularUser, related_name='regular_user', null = True)
    manager = models.ForeignKey(Manager, related_name='manager', null = True)
    status = models.CharField(max_length=30, choices=choices, default=read)
    reviewed_by = models.ForeignKey(Manager, related_name='reviewed_by', null = True)

    def __unicode__(self):
        return self.user.username
```

# System Design

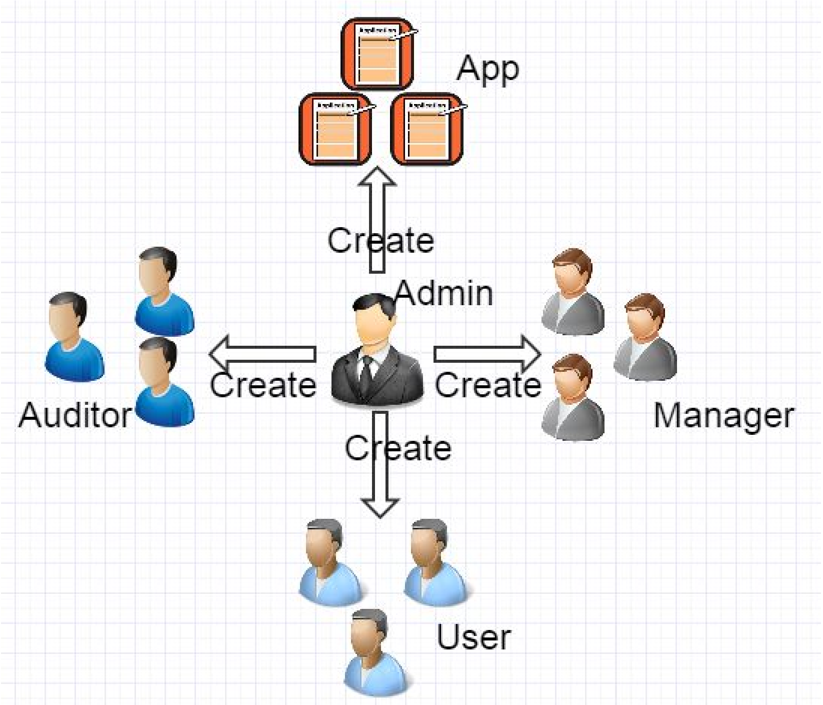
- Framework-Django

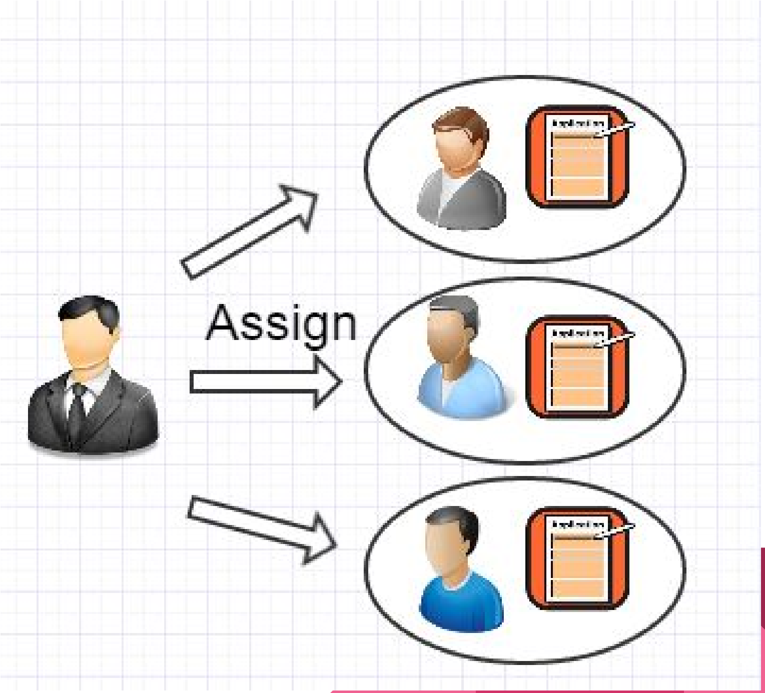
  Django is a high-level MVC Python Web framework that make web development quick and clean. Django is free and open source. It helps developers to  create complicated, database-driven websites.

- Database-Sqlite, PostgreSQL

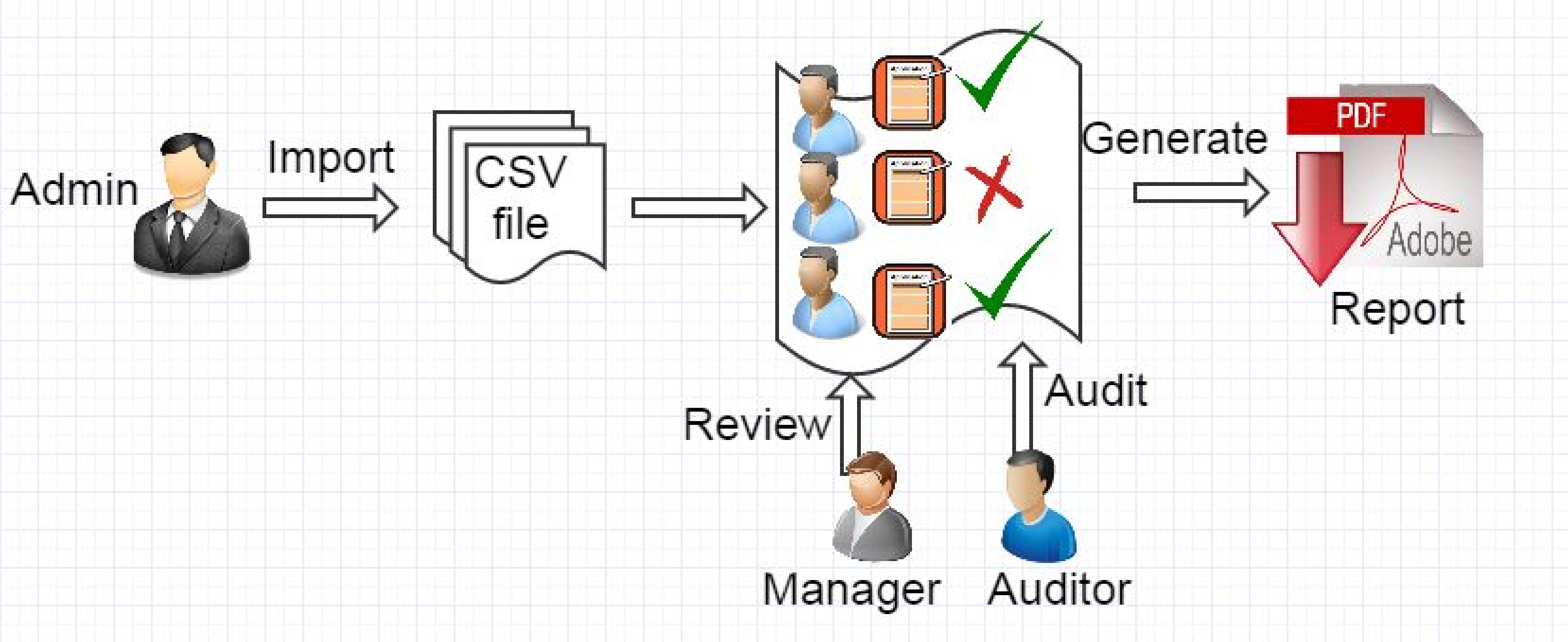
  Sqlite in development process, PostgreSQL in testing process.

# System Design-Workflow

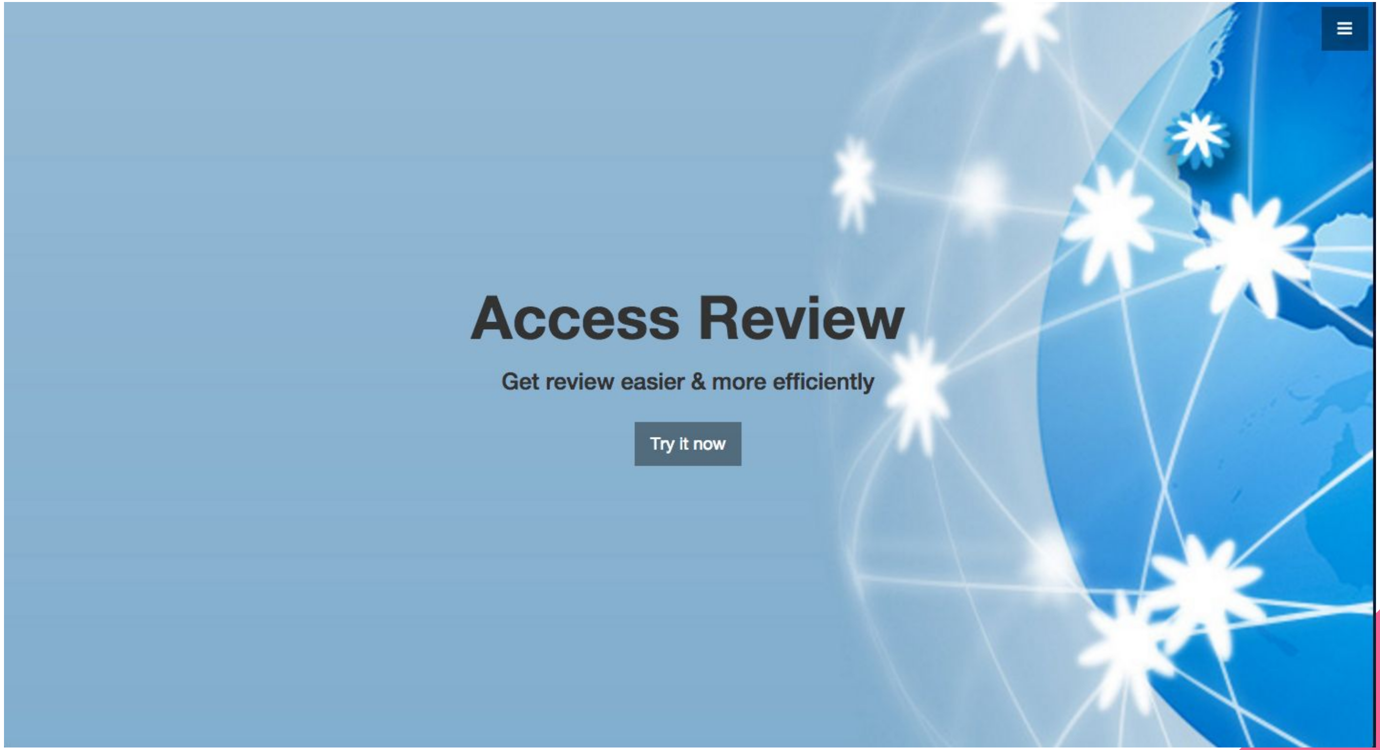




# System Design-Workflow



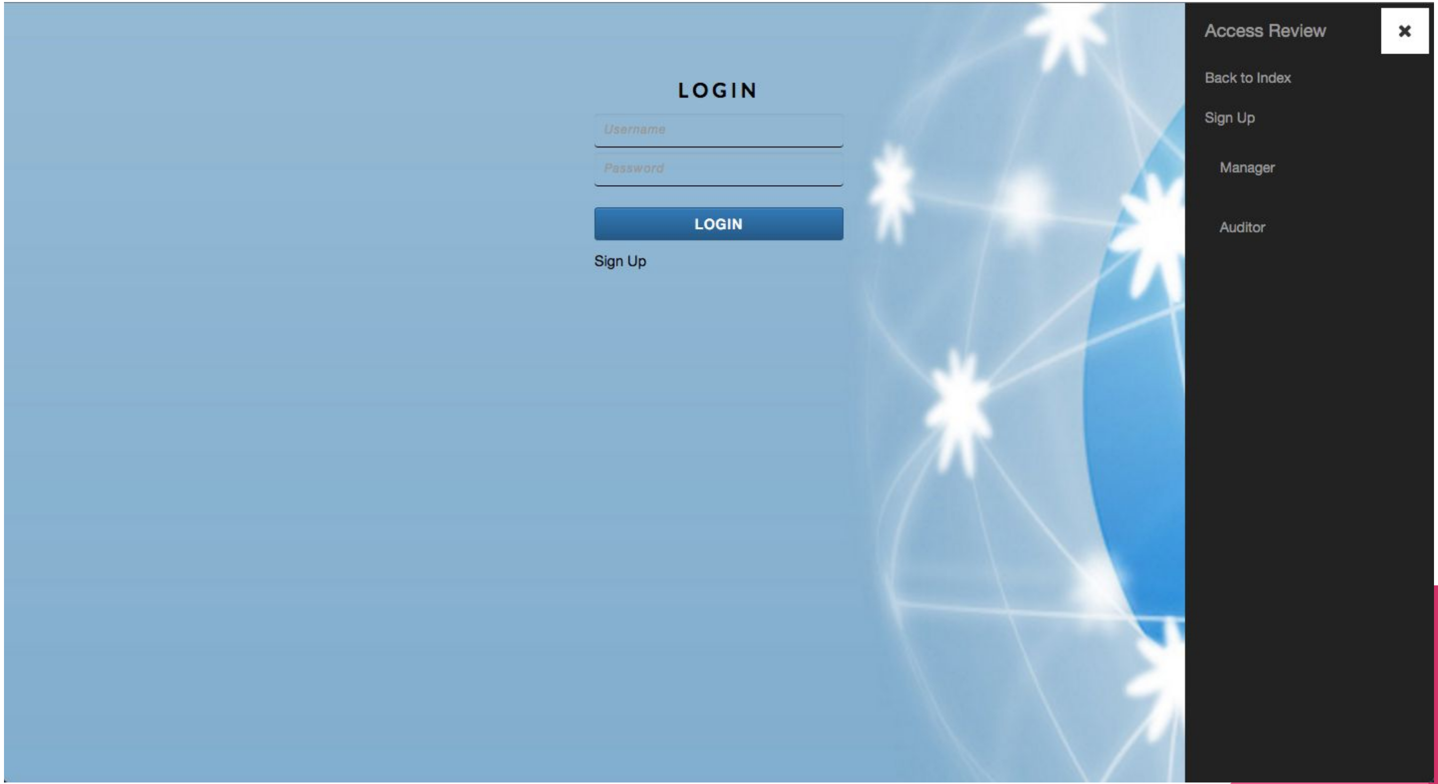
# Preliminary results



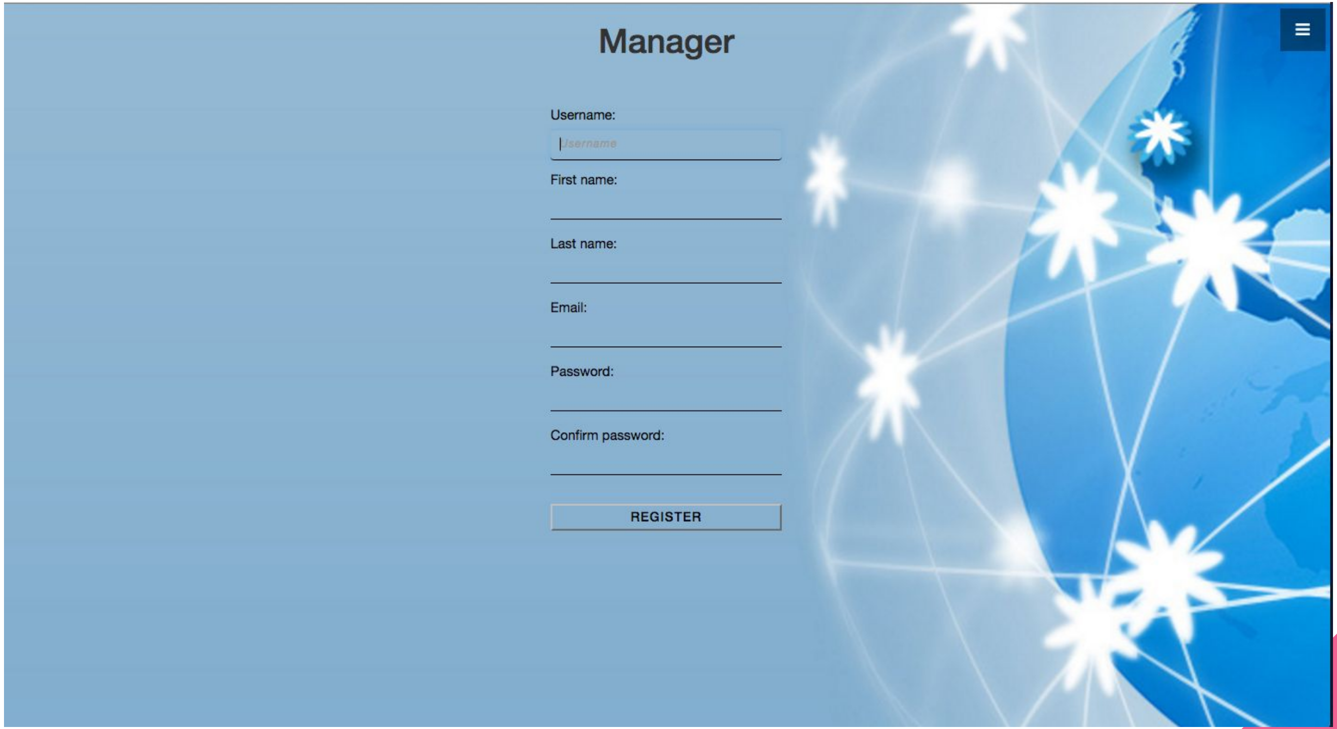
Access Review

Get review easier & more efficiently

# Preliminary results



# Preliminary results



Manager

Username:

Username

First name:

Last name:

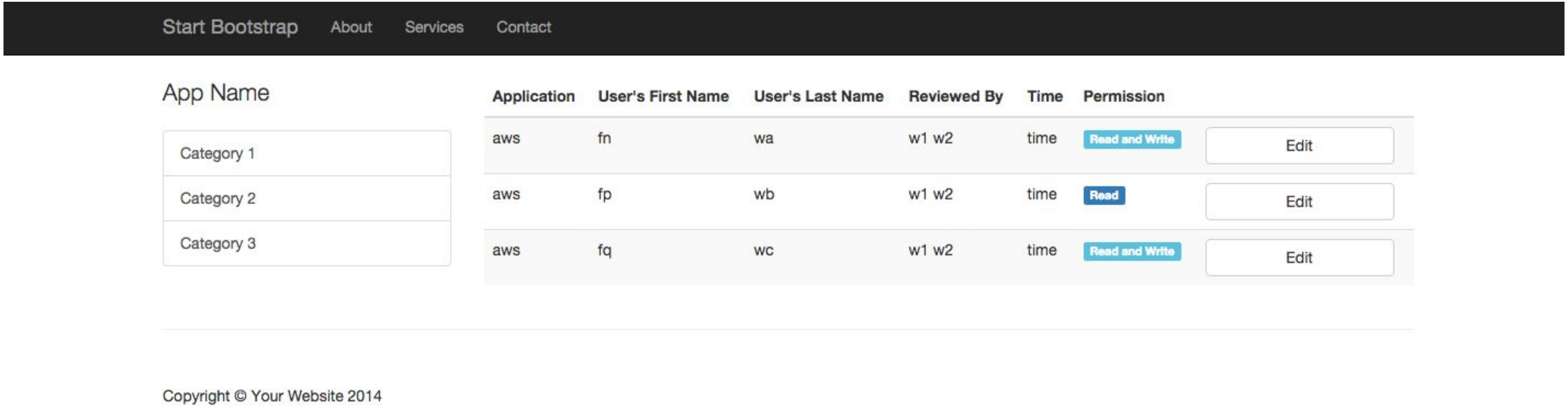
Email:

Password:

Confirm password:

REGISTER

# Preliminary results



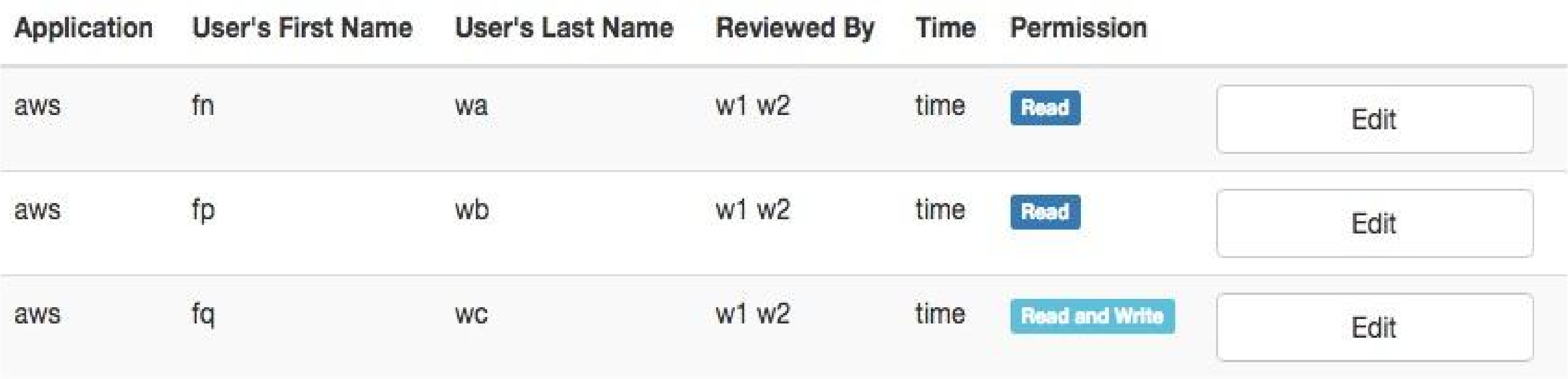
App Name

- Category 1
- Category 2
- Category 3

| Application | User's First Name | User's Last Name | Reviewed By | Time | Permission | |
|---|---|---|---|---|---|---|
| aws | fn | wa | w1 w2 | time | Read and Write | Edit |
| aws | fp | wb | w1 w2 | time | Read | Edit |
| aws | fq | wc | w1 w2 | time | Read and Write | Edit |

Copyright © Your Website 2014

# Preliminary results



| Application | User's First Name | User's Last Name | Reviewed By | Time | Permission | |
|---|---|---|---|---|---|---|
| aws | fn | wa | w1 w2 | time | Read | Edit |
| aws | fp | wb | w1 w2 | time | Read | Edit |
| aws | fq | wc | w1 w2 | time | Read and Write | Edit |

# Health of the project progress

- Development is on the right track as we expected.
- Still trying to refine web interface and backend logic.
- Creating more features to facilitate review process.

# Plan for the rest of project

## Iteration 3

| ID | Description | Progress |
|---|---|---|
| 3.1 | Improve the functions to show the review pages for auditors | Under construction |
| 3.2 | Refine the front end view to make it user-friendly | Under Construction |
| 3.3 | Refine database schema for better data normalization | Under Construction |
| 3.4 | Design the functions for administrator to assign the applications | Under Construction |
| 3.5 | Test for the application of this step | Under Construction |

# Plan for the rest of project

## Iteration 4

| ID | Description | Progress |
|---|---|---|
| 4.1 | Design the PDF format of the review report to download | Under Construction |
| 4.2 | Change the SQLite database into Postgresql database | Under Construction |
| 4.3 | Test the whole application for the usability and refine it | Under Construction |
| 4.4 | Test the whole application for integration and refine it | Under Construction |

# Reference

- Sandhu, Ravi S., and Pierangela Samarati. "Access control: principle and practice." *Communications Magazine, IEEE* 32.9 (1994): 40-48.
- Sandhu, Ravi S., et al. "Role-based access control models." *Computer* 2 (1996): 38-47.
- https://www.djangoproject.com/
- Repost from Richard Zorza’s Access to Justice Blog: New York Making Big Progress on Multi-Lingual Court Orders
- http://accessreview.ca/2015/10/19/repost-from-richard-zorzas-access-to-justice-blog-new-york-making-big-progress-on-multi-lingual-court-orders/
- https://www.digitalocean.com/community/tutorials/sqlite-vs-mysql-vs-postgresql-a-comparison-of-relational-database-management-systems
- http://jerel.co/blog/2015/07/an-introduction-to-emberjs-for-django-developers
- http://www.postgresql.org/
- https://en.wikipedia.org/wiki/Evaluation_approaches

# Questions