



Cloud Computing ASSIGNMENT 2 REPORT 2019

William Low: s3697652

Syed Hariz: s3701799

TABLE OF CONTENTS

Project details	3
Contribution	3
Links	3
Summary	4
Introduction	5
How does it work?	5
General User features	5
Author inherits General Users features	6
Website features	6
Why is this beneficial?	6
Who will use this?	7
Advantages of our website	7
Related work	8
Good Reads	8
LibraryThing	8
Bookish	8
Software Design/Architecture	9
Architecture	9
Front-end	10
Back-end	10
Data Structure	10
Authors	10
Books	11
Books has many categories relation	11
Books has many distributors relation	11
Categories	11
Distributors	11
Like book	12
Dislike book	12
Implementation	13

Developer Manual	13
Amazon Web Service account	13
Cloud9 AWS	13
Setting up AWS services	15
Amazon Simple Storage Service - S3	15
Amazon Relational Database Service - RDS	15
Amazon Simple Email Service - SES	15
Elastic Compute Cloud (Amazon 2) - EC2	15
Future works	16
User manual	17
General User features	17
Author features	19
References	21

PROJECT DETAILS

Contribution

William Low - 50%

Syed Hariz - 50%

Links

Website: ec2-18-237-115-211.us-west-2.compute.amazonaws.com

Project GitHub repository: https://github.com/liam619/cloud_project

SUMMARY

The objective of our project is to build a platform where booklovers can use it to receive detailed information of a book. This project was built with the idea of having a centralized website that feeds and returns an information of a book. Consequently, help future users using this website to explore their new favorite book to read or looking for a feedback about a particular book. This platform also acts as a 3rd party digital marketer to fellow book authors to expand the exposure of their book and help them receive a feedback of their book via the internet. The goal of this website is to provide reliable information to its users while having a high user experience and user interface.

INTRODUCTION

The purpose of this project was to create a centralized web service platform for book lovers to view and access the details of a book. The idea behind this project is to influence its user's point of view of a particular book, is it worth reading or worth scrolling through in the website's content. Given a situation where a user accesses the website, he/she can view the details of a book, the author of the book, description of the book and most importantly the review left for the book which are being expressed by like and dislike button. The target crowd of this website is not limited to users viewing an information of a book, we have decided to expand the crowd of the website by enabling its users to sign up as an author. Authors can create and publish their book on our website which simultaneously feed the website's data in real time. This particular type of project was built on top of Amazon Web Services (AWS). AWS platform provides reliable, scalable and inexpensive cloud computing services which proves to be beneficial towards developing our project. However, we have decided to narrow down its services and use which are only relevant towards the deployment of our project. In this report, we will cover important details of the website from front-end to back-end view for future users and developers to use and build respectively.

How does it work?

Alike many websites that can be surfed through the internet, our website can be accessed through the link attached on this report. Users can access the website a click away from the browser of their choice. One of the objectives developing our project is to create a user-friendly website. Thus, we have created a simple and direct features with minimalistic theme website. The website's features can be broken down into 2 point of views; General User and Author.

General User features

- Browse all books - Displaying all books that were issued in the website
- Book - Displaying the details of a book being viewed in the website
- Category - Browsing through a specific type of book category
- Distributor - Browsing through a specific type of book distributor

Author inherits General Users features

- *Browse all books*
- *Book*
 - Edit the details of the book
- *Category*
 - Create new category
- *Distributor*
 - Create new distributor
- New Book - Creating a new book to be published on the website
 - Name - Name of the book
 - Description - Description of the book
 - Categories - Category of the book (Eg. Fantasy, Sci-fi)
 - Distributors - Distributor of the book (Eg. Pearson)
 - Image - Image of the book
- Profile - Viewing and Editing user's information
 - Name - Name of user
 - Email - Change to a new email address
 - Password - Change to a new password
 - Confirmation - New password confirmation

Website features

- Sign up - User signing up as an Author
- Log in - User logging in as an Author
- Contact Us - User contacting website's customer service via email

Why is this beneficial?

Our website can help our users in many different ways. Our website can assist users before purchasing a new book. Users can view varieties of book and narrow it down to their likings by browsing through a specific type of book category or their favorite distributors. Users can refer to our platform to investigate the feedback from other booklovers about a particular book by inspecting at how many likes or dislikes the book has received. Authors can publish their new released books through our website which can help them to market their product to a large amount of crowd, free of charge.

In terms of cloud platform used, the benefits of using Amazon Web Services and their services are we can quickly and easily configure these services based on our requirements and needs. Scalable hosting, large storage capacity, multiple zones of deployment, services that prevent us from reinventing, creating or using different and multiple source of API's for each services our website can offer. It is better to have all services that are being used by our website in a centralized unit to enhance the manageability and scalability of our project.

Who will use this?

Our target crowd are specifically towards the people with an interest of reading books and people that are looking to publish their books on the internet.

Advantages of our website

- Book lovers can explore varieties of book before making a purchase at their local bookstore or online
- Book lovers can discover their next favorite book
- Book lovers can leave a positive or negative feedback of the book
- Authors can publish their new book to a vast internet crowd
- New authors can publish their book
- Authors can view the feedback of their book

RELATED WORK

Good Reads

Goodreads is a social cataloging website that allows individuals to freely search its database of books, annotations, and reviews. Users can sign up and register books to generate library catalogs and reading lists. They can also create their own groups of book suggestions, surveys, polls, blogs, and discussions.

Link: www.goodreads.com

LibraryThing

LibraryThing is a social cataloging web application for storing and sharing book catalogs and various types of book metadata. It is used by authors, individuals, libraries, and publishers.

Link: www.librarything.com

Bookish

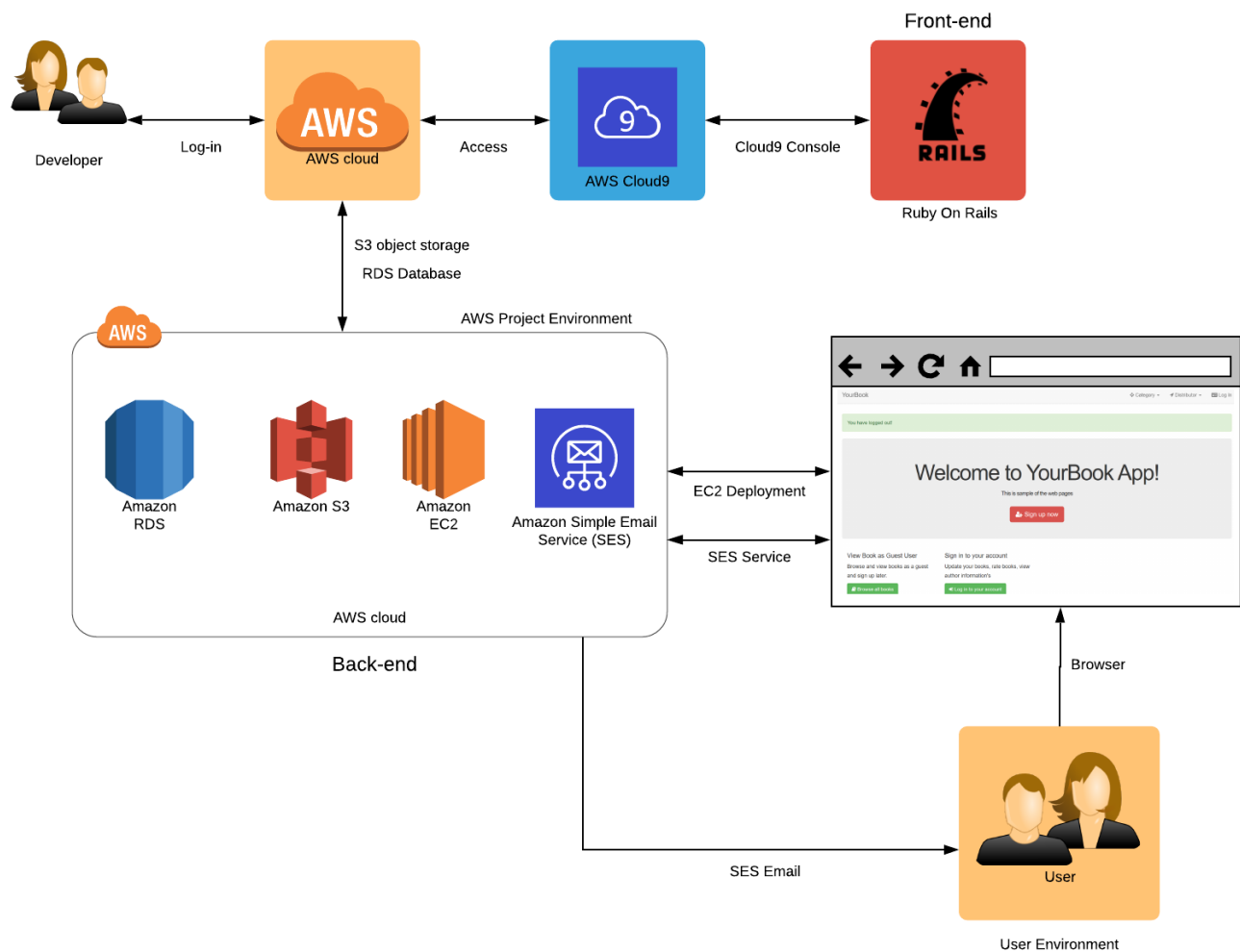
Bookish is a content discovery and ecommerce website. Bookish is an editorial content site connecting readers with books and authors. The goal of the website is to help readers discover their next favorite book and to help book clubs discover their next read.

Link: www.bookish.com

SOFTWARE DESIGN/ARCHITECTURE

A high level overview of the architecture of the project. This is regarded as high level as there are many different elements and services that were added into this project. The overview covers from the platform that are being used to create the front-end of the website and the back-end services that were embedded onto the project.

Architecture



Front-end





- Ruby on Rails
 - API ready
 - HTML
 - CSS
 - Bootstrap
 - Ruby

Back-end





- Amazon RDS
 - Relational Database service in the cloud
- Amazon S3
 - Object storage service
- Amazon EC2
 - Web service and deployment, running virtual servers
- Amazon SES
 - Cloud-based email sending service

Data Structure



Authors

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	name	varchar							NULL
3	email	varchar							NULL
4	password_digest	varchar							NULL
5	remember_digest	varchar							NULL
6	admin	boolean							0
7	created_at	datetime							NULL
8	updated_at	datetime							NULL



Books

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	name	varchar							NULL
3	author_id	integer							NULL
4	description	text							NULL
5	image	varchar							NULL
6	created_at	datetime							NULL
7	updated_at	datetime							NULL





Books has many categories relation

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	book_id	integer							NULL
3	category_id	integer							NULL





Books has many distributors relation

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	book_id	integer							NULL
3	distributor_id	integer							NULL





Categories

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	name	varchar							NULL
3	created_at	datetime							NULL
4	updated_at	datetime							NULL





Distributors

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	name	varchar							NULL
3	created_at	datetime							NULL
4	updated_at	datetime							NULL

Like book

	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	book_id	integer							NULL
3	like_count	integer							1
4	created_at	datetime							NULL
5	updated_at	datetime							NULL

Dislike book

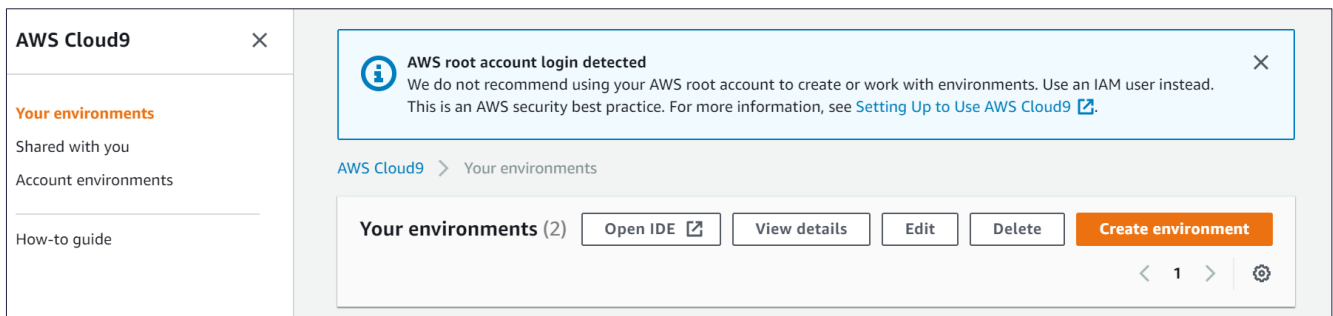
	Name	Data type	Primary Key	Foreign Key	Unique	Check	Not NULL	Collate	Default value
1	id	integer							NULL
2	book_id	integer							NULL
3	dislike_count	integer							1
4	created_at	datetime							NULL
5	updated_at	datetime							NULL

IMPLEMENTATION

Developer Manual

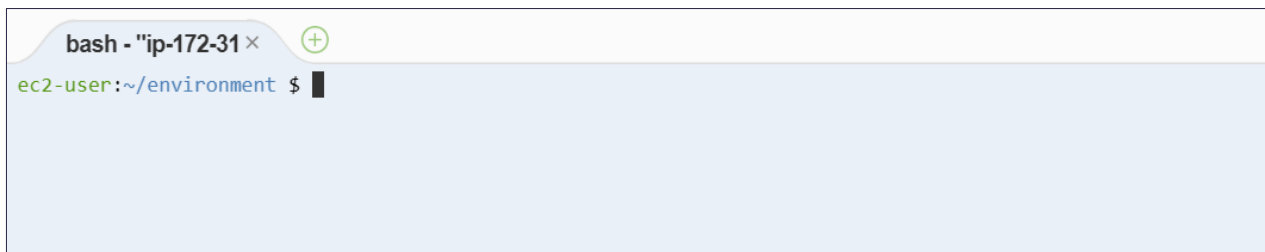
Amazon Web Service account

Setting up the project is pretty straight forward. Log in to Amazon Web Services account if not sign up an account. Once you are in the AWS Management Console, select Cloud9 AWS service then select your environment (any will do), if not create a new one and click on the Open IDE button. Example of what will be prompted once you are in Cloud9 AWS;

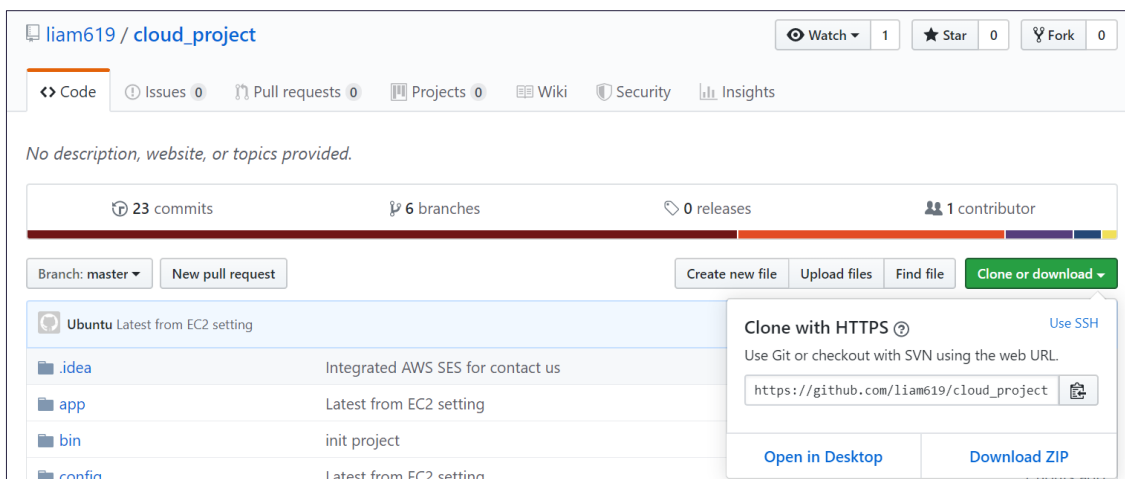


Cloud9 AWS

From there on, you will see a terminal at the center of the environment as shown below;



Open a new tab and go to the GitHub link that was provided in the report, click on the green button that says “Clone or download”. Click on it and it will display a link, copy the link.



Once copied, go back to your Cloud9 tab and in the terminal run the command as follows;

```
bash - "ip-172-31" × (+)  
ec2-user:~/environment $ git clone https://github.com/liam619/cloud_project.git
```

Hit enter and wait until the process is done. This command will clone all the project's front-end code into your Cloud9 environment. Once completed, you should be able to access the project's folder in the dashboard on the left as "cloud_project". Knowledge of Model View Controller, Ruby on Rails, HTML, CSS and usage of Bootstrap are essential prior to developing the project. If not, search up "Ruby on Rails tutorial by Michael Hartl" to understand required knowledge that is essential for the project.

From the environment directory, cd into the project directory by running this command;

```
bash - "ip-172-31" × (+)  
ec2-user:~/environment $ cd cloud_project
```

After you are in the project folder directory, it is important that you run the following steps;

Step 1 - Install the gems

```
bash - "ip-172-31" × (+)  
ec2-user:~/environment/cloud_project (master) $ bundle install
```

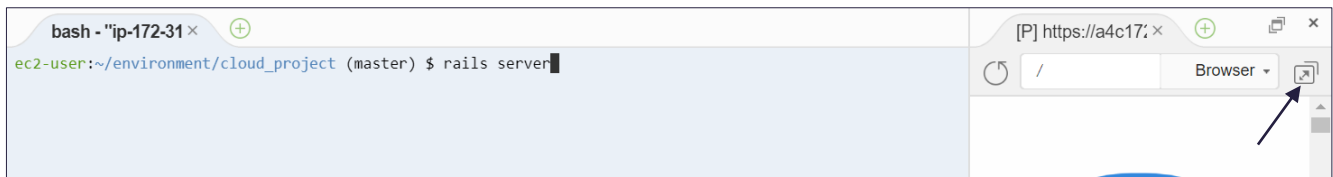
Step 2 - Migrate the database

```
bash - "ip-172-31" × (+)  
ec2-user:~/environment/cloud_project (master) $ rails db:migrate
```

Step 3 - Populate data in database

```
bash - "ip-172-31" × (+)  
ec2-user:~/environment/cloud_project (master) $ rails db:seed
```

Finally, to run the website as local host, run this following command;



Once the Puma server is running, you can view the website by clicking on the button with an arrow direction pointed at it in the picture above. To stop the server from running hit Ctrl + C in the terminal.

Setting up AWS services

Amazon Simple Storage Service - S3

Amazon Simple Storage Service is storage for the Internet. It is designed to make web-scale computing easier for developers. Amazon S3 has a simple web services interface that you can use to store and retrieve any amount of data, at any time, from anywhere on the web. Refer to the link on how to set up S3 for the website. (1)

Amazon Relational Database Service – RDS

Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database in the cloud. Here are the step by step guide on setting RDS for the website. (2)

Amazon Simple Email Service – SES

Since we require an email service for our users to contact us, it is essential for the website to provide such feature. Amazon Simple Email Service (SES) is a cloud-based email sending service designed to help digital marketers and application developers send marketing, notification, and transactional emails. To set up SES in the website follow this guide. (3)

Elastic Compute Cloud (Amazon 2) – EC2

Since our front-end is being developed with Ruby on Rails, setting up EC2 in our project is much easier just by running a few line of commands in the Cloud9 terminal. Guide to set up EC2 for Ruby on Rails project. (4)

Future works

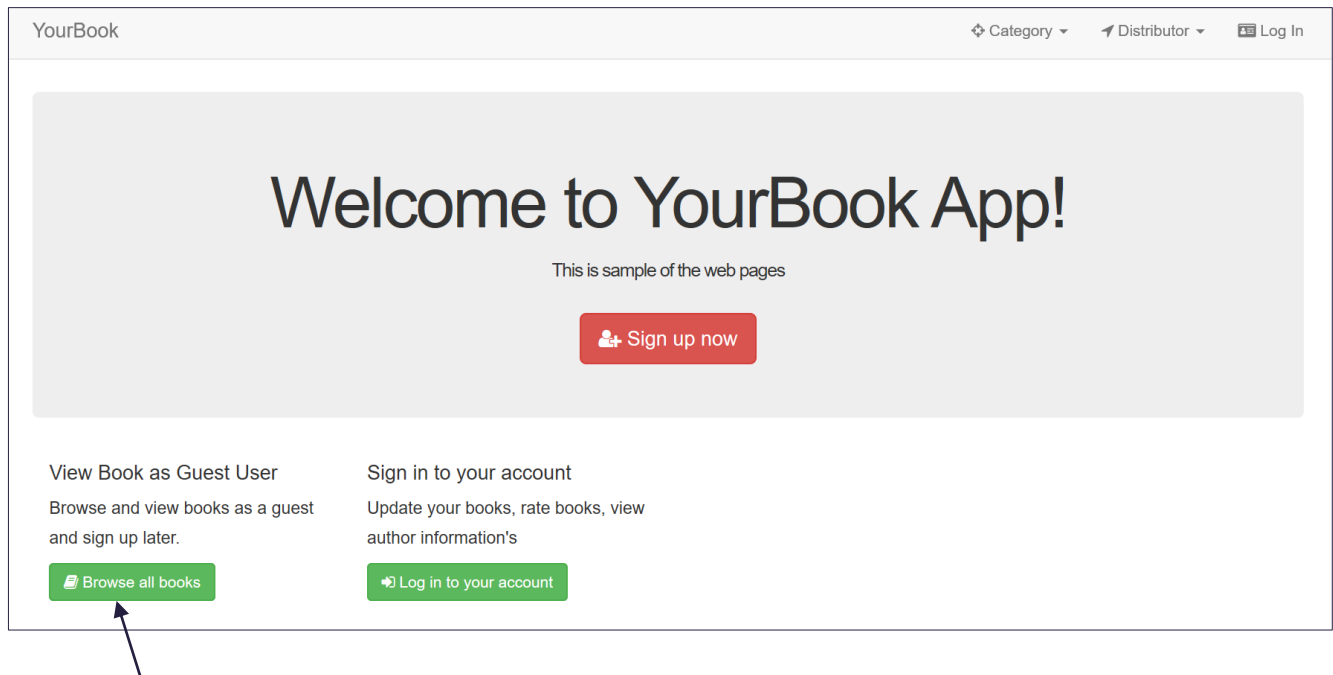
This project is still in its early stages. There are additional features worth considering that can help enhancing the website's features and usability;

- A comment box which enables the users to leave comments about the book
- Author will be notified once any feedbacks or comments are made about their book
- Ranking system in the website displaying top-rated books
- Appropriate logo for the website
- Additional information of the book where the books can be purchased online

USER MANUAL

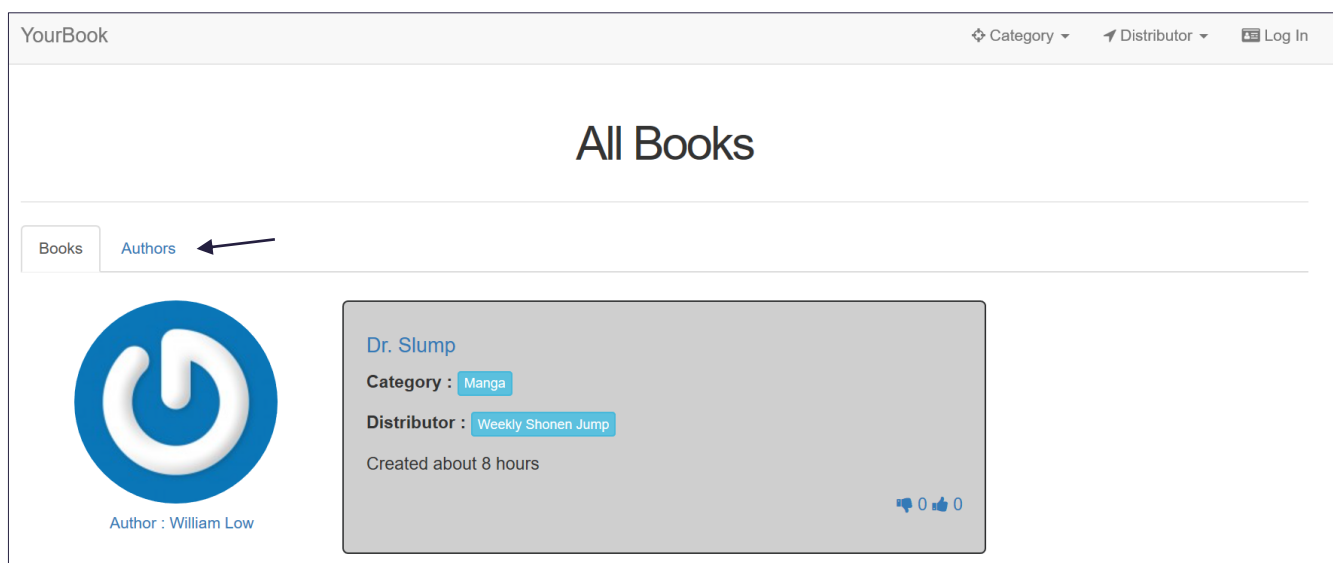
A step by step guide and elaboration of how users can access the features of the website.

Launch your ideal web browser (Eg. Google chrome, Internet Explorer). Copy and paste the website link that was attached on this report and you will be prompted the Home page of the website;



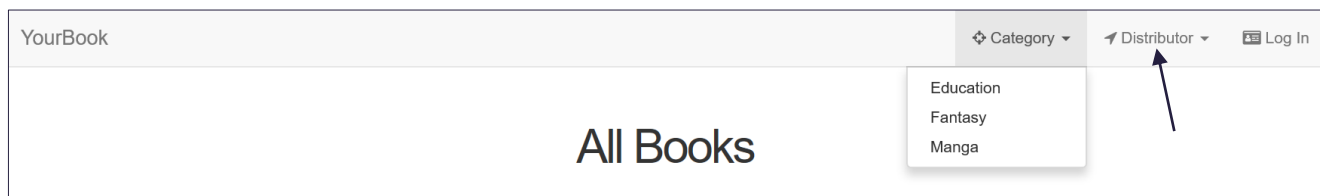
General User features

For general users to view all of the books that were published on the website, they can click on the “Browse all books” button below the Home page of the website as being displayed above.



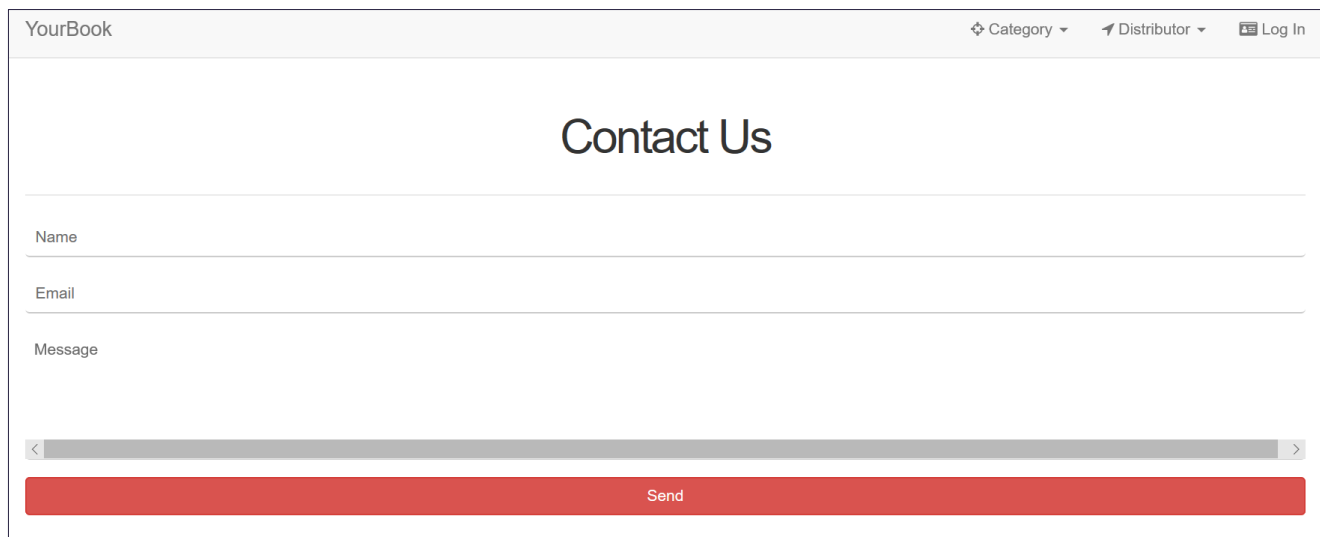
Users can click on any book they find interesting and view the details of the book and also leaving a like or dislike according to their judgment about the book. Users can also view the authors that are registered in the website by clicking on the Authors tab above.

Users can filter out the books according to their desired category or “genre” by clicking on the Category button on the navigation bar;



Users can also view all published book by their favorite distributors by clicking on the Distributor button on the navigation bar as shown in the picture above.

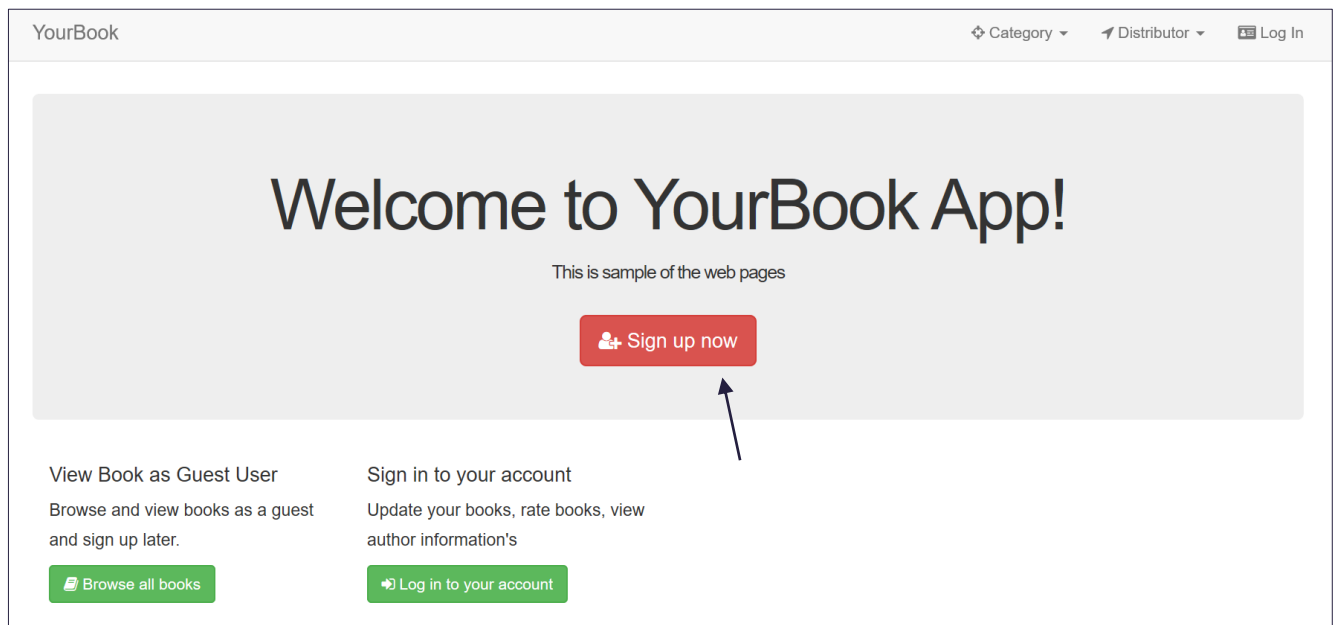
User can contact our team by sending an email through Contact Us page;

A screenshot of a web application interface showing the 'Contact Us' page. The navigation bar at the top is identical to the previous screenshot. The main content area has 'Contact Us' in a large, centered font. Below the title, there is a contact form with three input fields: 'Name', 'Email', and 'Message'. At the bottom of the form is a red button labeled 'Send'. A horizontal scrollbar is visible below the 'Message' field.

Link to Contact Us page can be found in the footer of every page in the website.

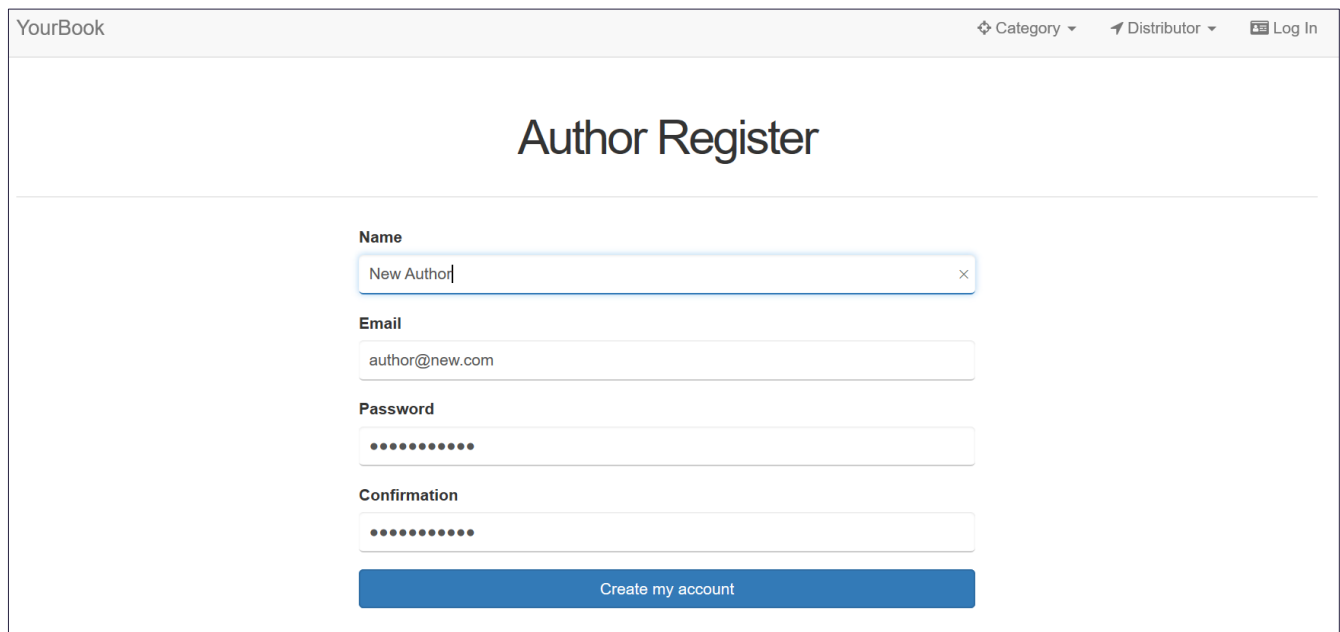
Author features

Signing up is required for authors to access the extended features of the application. In the Home page of the website, authors can click on the big red button that says “Sign up now”;



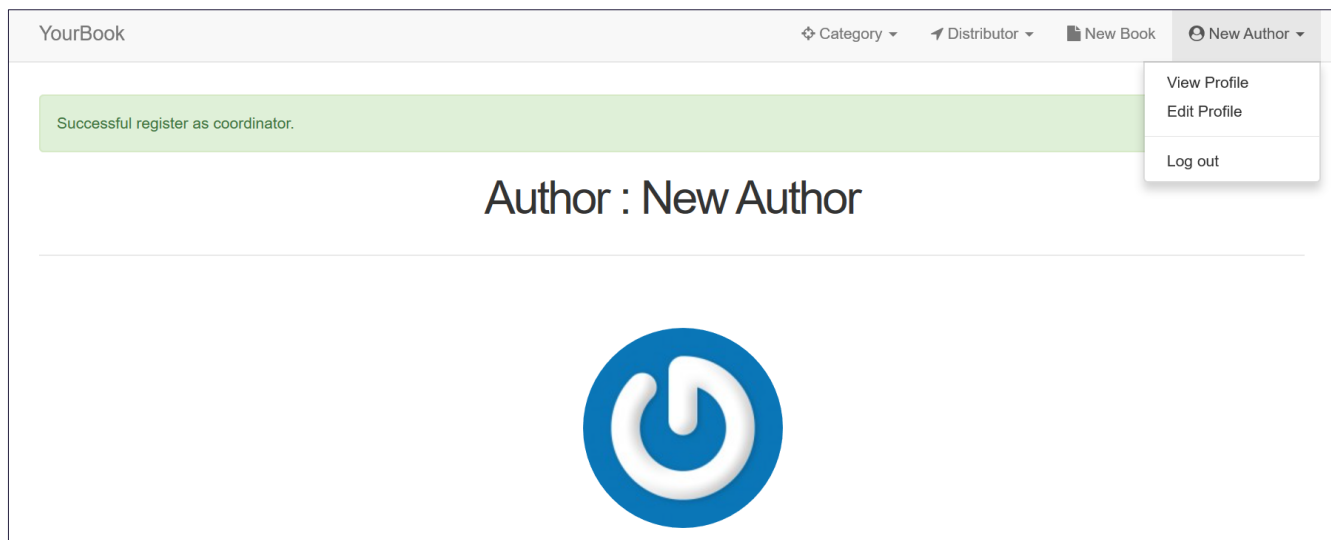
The screenshot shows the home page of the 'YourBook' application. At the top, there is a navigation bar with the 'YourBook' logo on the left and links for 'Category', 'Distributor', and 'Log In' on the right. The main content area has a large grey background with the text 'Welcome to YourBook App!' in the center. Below this, a smaller text says 'This is sample of the web pages'. A prominent red button with a user icon and the text 'Sign up now' is centered. A blue arrow points to this button. Below the main content, there are two columns of options. The left column is titled 'View Book as Guest User' and describes browsing as a guest, with a green 'Browse all books' button. The right column is titled 'Sign in to your account' and describes updating books and viewing author information, with a green 'Log in to your account' button.

This will take the author to a new page as being shown below;



The screenshot shows the 'Author Register' page. It has the same navigation bar as the home page. The title 'Author Register' is centered at the top. Below the title, there are four input fields: 'Name' (containing 'New Author'), 'Email' (containing 'author@new.com'), 'Password' (masked with dots), and 'Confirmation' (masked with dots). At the bottom, there is a blue button labeled 'Create my account'.

Once an account is created, author can view their profile or edit the details of their profile by clicking on their name on the top right hand-side of the website;



Author can change their name, change their current email to a new email and change their password to a new one in Edit Profile page.

A screenshot of the 'Edit Profile' page in the 'YourBook' application. The navigation bar at the top is identical to the previous screenshot. The main heading is 'Edit Profile'. Below the heading, there are four form fields arranged vertically. The first field is labeled 'Name' and contains the text 'New Author'. The second field is labeled 'Email' and contains the text 'author@new.com'. The third field is labeled 'Password' and contains the placeholder text 'Must be at least 8 alphanumeric with symbol'. The fourth field is labeled 'Confirmation' and is currently empty. At the bottom of the form, there is a blue button with the text 'Update Profile Information'.

REFERENCES

- 1 Amazon. Setting up S3. [Online]. Available from: <https://aws.amazon.com/s3/getting-started/>.
- 2 Service AW. Create an RDS DB Instance. [Online]. Available from: https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/CHAP_Tutorials.WebServerDB.CreateDBInstance.html?fbclid=IwAR3XVdftOBo6-ZdPgVCxJEjk_5WqMF_gkOdh2ytoAdC2WWR1-FRnwg4gXw.
- 3 Amazon. Setting up Email with Amazon SES. [Online]. Available from: <https://docs.aws.amazon.com/ses/latest/DeveloperGuide/setting-up-email.html>.
- 4 StackOverflow. Setting up EC2 on Ruby on Rails. [Online]. Available from: <https://stackoverflow.com/questions/50129155/deploying-a-rails-5-app-to-ec2-rds-db-instance-connectivity-error?fbclid=IwAR1uVf0jGsQ74OcwwmwEp9DStl-M7yCPuNtMf5sU6bfuOGiTaq1dMMiazGQ>.