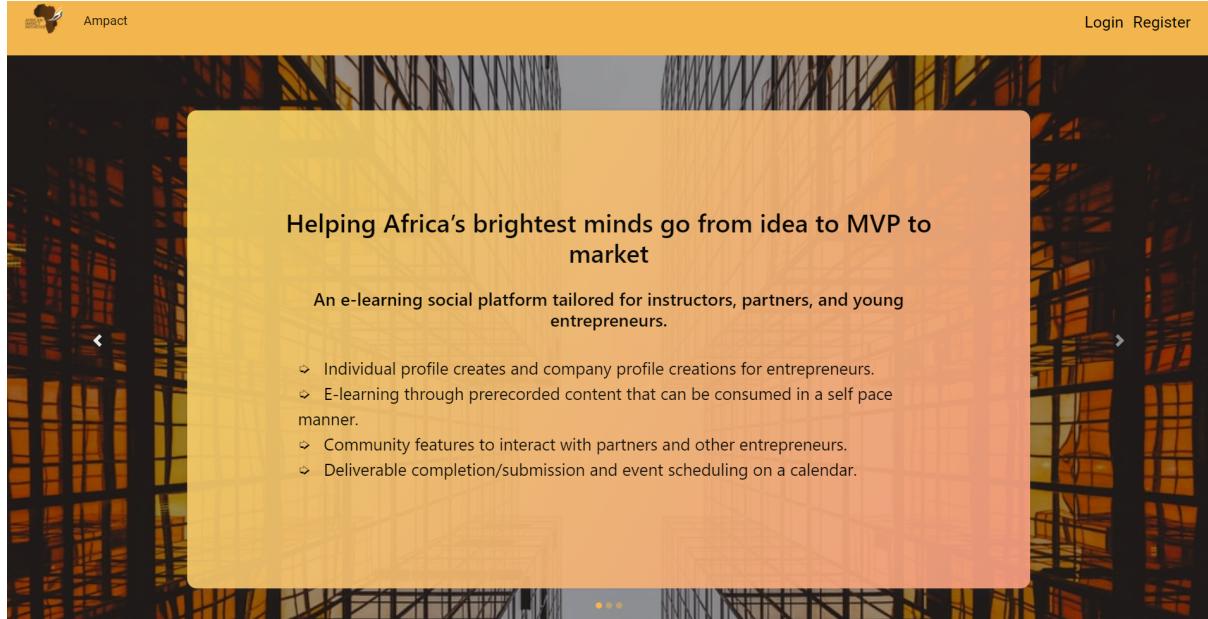


Frontend React Components & Pages Documentation

Components

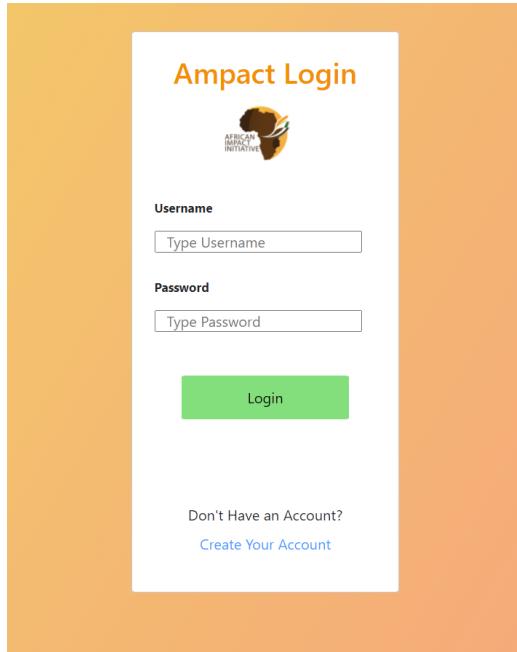
1. Pages

- [Landing.js](#)



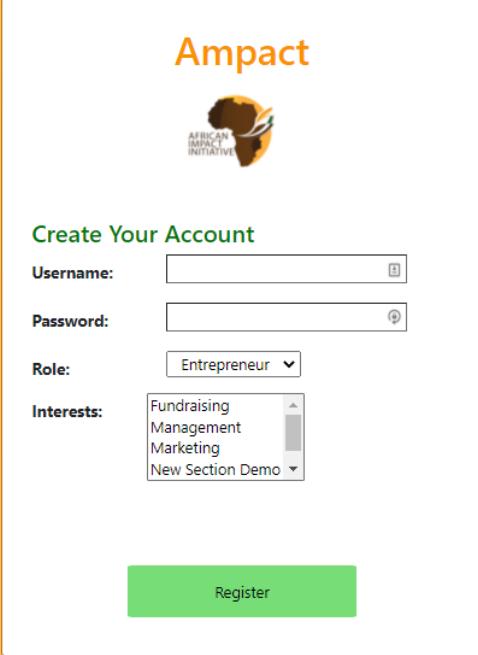
Landing.js is the landing page React component where users are directed to when they first enter the website. There is a carousel which displays various information about our platform such as an about me, the browsing and group creation feature.

- [Login.js](#)



A component that renders the login page. The user can enter their username and password. An error will appear if the login has failed.

- **Registration.js**



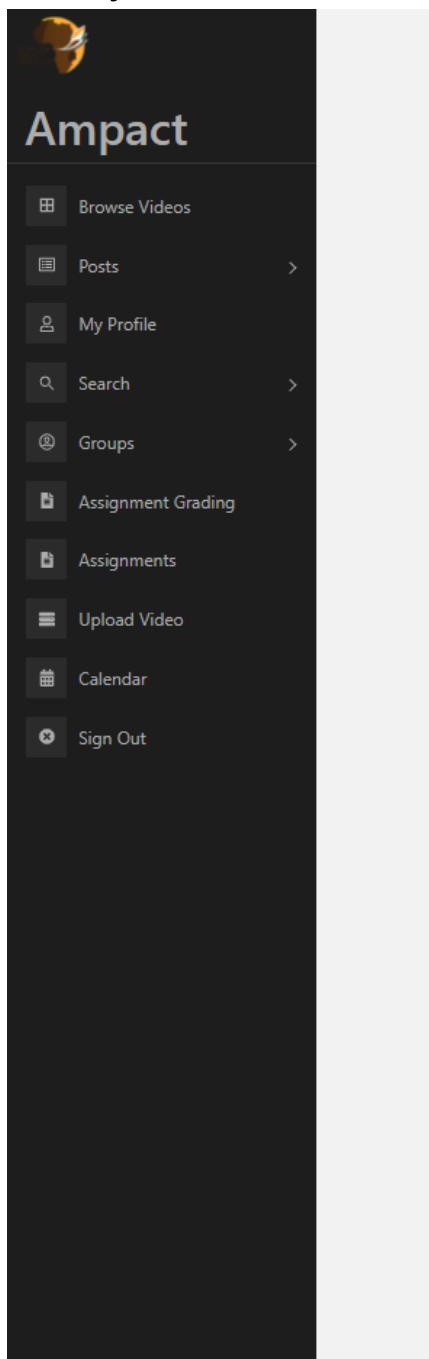
The image shows a registration form titled "Create Your Account" for the "Ampact" platform. The form is set against a white background with a green header and footer. At the top, there is a logo for "AFRICAN IMPACT INITIATIVE" featuring a stylized map of Africa. Below the logo, the word "Ampact" is written in a large, orange, sans-serif font. The form fields include:

- Username:** A text input field.
- Password:** A text input field with a visibility icon.
- Role:** A dropdown menu currently set to "Entrepreneur".
- Interests:** A dropdown menu containing the following options:
 - Fundraising
 - Management
 - Marketing
 - New Section Demo

At the bottom of the form is a green "Register" button.

A component that renders the registration page. The user can enter the user, password, ,role and select their interests. An error will appear if the fields are not specified and the username exists.

- **NavBar.js**



This component shows the navigation bar when the user is logged in. You can navigate to important parts of the website and sign out.

- **GroupProfile.js**



DoorWatchers

MemberID: Add member Remove Member Edit

About

Smart home video doorbell to secure your home.

Group Members

ddd

Shows a profile for the page (About, and group members). The user can add/remove a member.

- **GroupProfileEdit.js**

Edit Group Information

Group Name

This is the name of your group or company.

Group Picture

Group Description

UpdateDelete Group

This component renders the page to edit a group. The user can change the picture, about, and name.

- **GroupProfileCreate.js**

Create New Group

Group Name

This is the name of your group or company.

Group Picture

Group Description
 G

Submit

Groups

Search for groups **Search**

[!\[\]\(4fc2f7cc81639d6310529267a2ee602c_img.jpg\) DrawIO](#)
A mobile drawing app that allows users to draw using AI based assistance.

[!\[\]\(714d70875eaf0e2f34d0a261eaf96dad_img.jpg\) AirBudsMax](#)
The only wireless earbuds you will ever need.

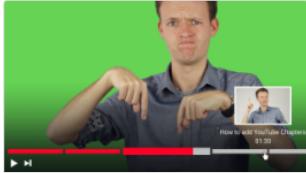
[!\[\]\(dc7d17b015a4a5f15a29473bc04652a8_img.jpg\) Emerald Corp. Investing](#)
We are a investing advisory council for business'.

Renders the group creation page. The user can view other groups and create their own group. Also you can search for groups on the same page. You can search based on group name. The searching is not case sensitive.

- **Browse.js**

Browse

Recommended



How to Get More View With Tags

Wondering which YouTube tags to use to rank videos in YouTube and Google? To leverage YouTube tags to their fullest potential, check out these tips and tricks.



YouTube Titles > With These Words Your Videos Get Clicked

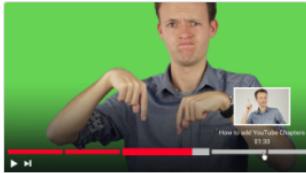
You're asking how to write YouTube titles that get views? This video explains how to write YouTube video titles that show great results regarding Click-Through-Rate and Video SEO.



How to Write a Description for YOUTUBE > TIPS and EXAMPLES

The complete guide on how to write a description for YouTube. YouTube descriptions can boost SEO, subscriptions, view counts, and watch time. They can also help your videos rank on YouTube.

Marketing



How to Get More View With Tags

Wondering which YouTube tags to use to rank videos in YouTube and Google? To



YouTube Titles > With These Words Your Videos Get Clicked

You're asking how to write YouTube titles that get views? This video explains how

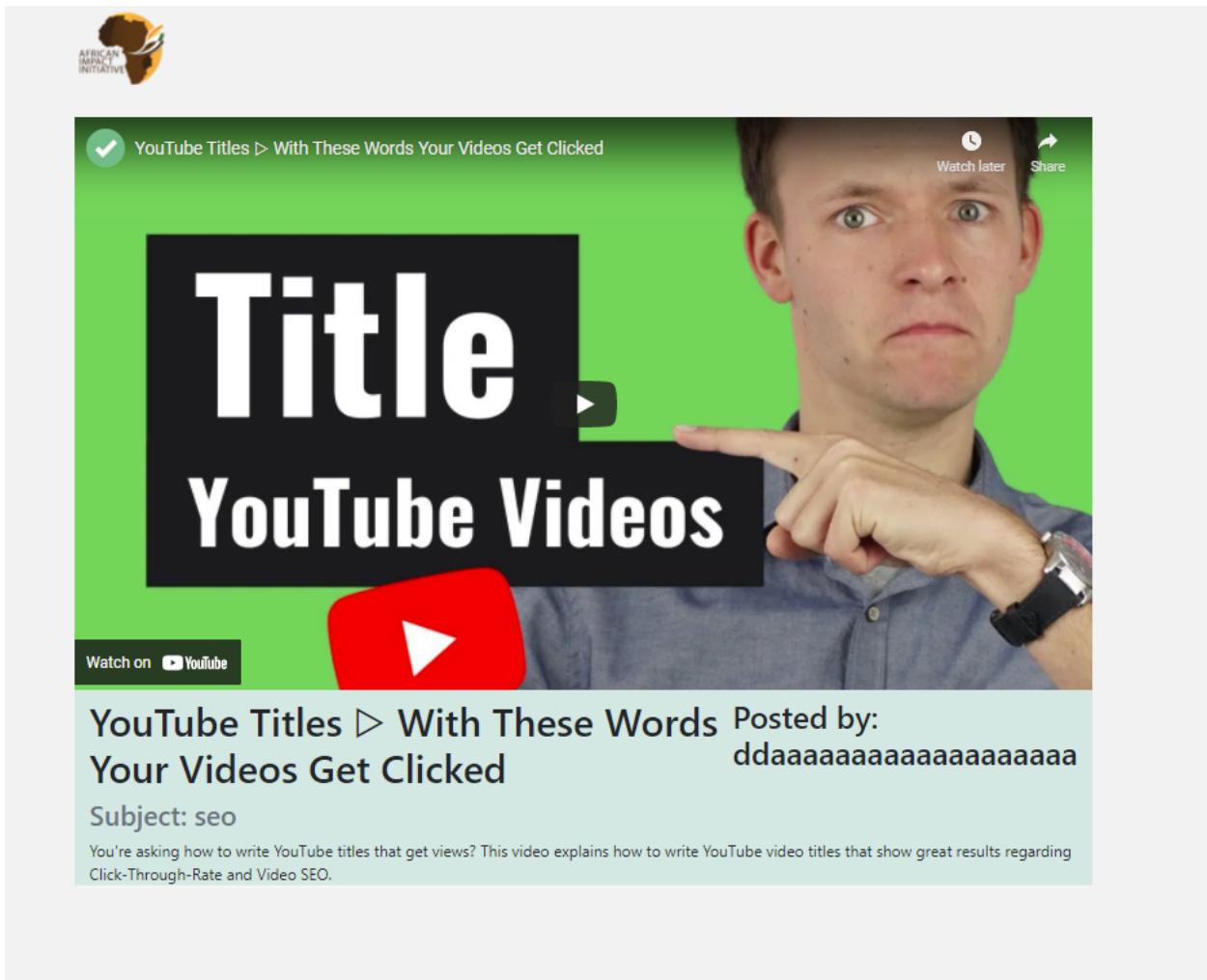


How to Write a Description for YOUTUBE > TIPS and EXAMPLES

The complete guide on how to write a

Renders the browse video page. Users can view all the videos uploaded by instructors. Users also have a recommended videos section that is generated based on their interests.

- **SingleVideoPage.js**



Renders the single video page. Users after browsing and choosing their videos can view the embedded youtube video which includes information about the video title, poster, subject and its description.

- **SingleVideoAdd.js**

Upload New Video

Title

Enter the title here

This will be the display title.

Description

Video Thumbnail

Enter Thumbnail link

Video Link

Enter Youtube video link

Should be a Youtube embed video link.

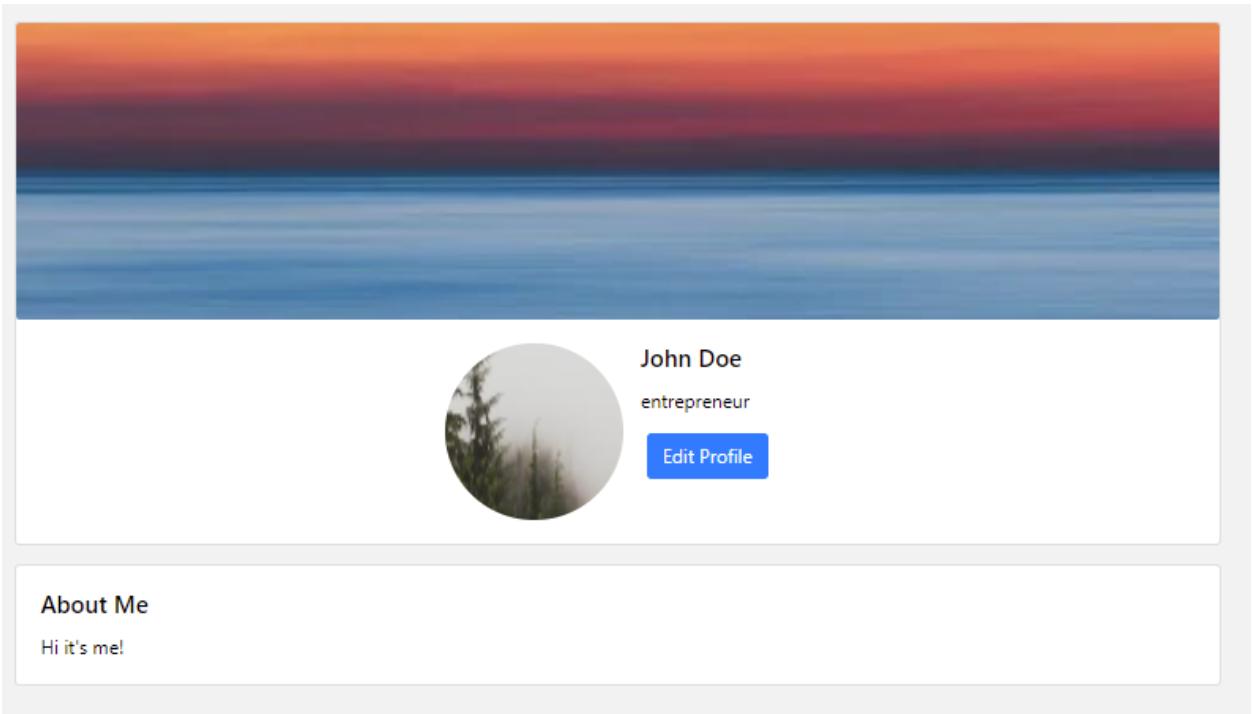
Tag

Enter video tag

Tags determine video section display.

Renders the single video add page. When uploading you can specify the title, description, video thumbnail, video link and a tag. After doing so you can upload the video

- **UserProfile.js**



Renders the user profile page with a navigation bar. It displays the user's First name, last name (or username if it isn't changed), the user role, an about me, a profile picture and a default banner photo.

- **UserProfileEdit.js**

Edit User Profile

First Name

Last Name

Profile Picture

About

Update

Renders the user profile edit page. In it, you can fill out your first name, last name, profile picture and about me which after hitting update will update your user profile.

- **SearchUserProfiles.js**

User Profiles

A screenshot of a web application showing search results for user profiles. At the top, there is a search bar with the placeholder "Search for user profiles" and a text input field containing "david". To the right of the input field is a blue "Search" button. Below the search bar, there are navigation buttons for navigating through the results: two small arrows pointing left and one large arrow pointing right. The main area displays three user profiles in cards:

- David Kelly**
David Kelly
- David Russ**
David Russ
- David Tan**
David Tan

Renders the User profile searching page. It renders all the users that match your query. It defaults to showing everyone if you haven't searched or searched blank. You can type their first or last name to search for users. The searching is not case sensitive.

- **CalendarPage.js**

A screenshot of a calendar application for July 2021. The calendar grid shows days from Sunday to Saturday. Some days have specific events marked:

- Wednesday, July 14:** A yellow box highlights the entire day, with a tooltip "2a personal meeting" appearing over the bottom right corner.
- Thursday, July 20:** A red dot is placed on the 20th, with a tooltip "2a general meeting" appearing over the 19th.

The top of the calendar has a header with "Add Event", the month "July 2021", and navigation buttons for "Today", "Month", "Week", "Day", and arrows for "Previous" and "Next".

Renders the calendar page of the website. Users can change the view of the calendar to day, week, or month. Uses the [FullCalendar API](#).

- **PostFeed.js**

Discussion

Create Post As: christophersuh

Type your post here!

Set visibility:

All

Submit

 christophersuh
Type: QnA 2021/07/15
Visible to: all
hi

Comments

 davidtan
Type: QnA 2021/07/14
Visible to: all
1

Comments

 davidtan
Type: QnA 2021/07/14
Visible to: all
1

Renders all the posts on the discussion (QnA) board by visibility and also allows the creation of new posts.

- **PostFeedAnnoucements.js**

Annoucements

Create Post As: davidtan

Type your post here!

Set visibility:

All

Submit

**christophersuh**

Type: announcement 2021/07/15
Visible to: all
ssdsdasdasd

**christophersuh**

Type: announcement 2021/07/15
Visible to: all
delete this

**davidtan**

Type: announcement 2021/07/15
Visible to: partner
ANNOUCMENT FROM ENTRE TO PARTNER

Renders all the posts on the announcement (announcement) board by visibility and also allows the creation of new posts on the announcement board.

- **DeliverableFeed.js**

Assignments

[Create New Assignment](#)

Tets assignmnet 1

Due: 7/31/2021 10:09:10 PM TotalMarks: 12

Date Created: 7/30/2021 1:55:36 PM

[Submissions](#)

[Details](#)

final assignment test

Due: 7/10/2021 9:58:28 PM TotalMarks: 12

Date Created: 7/29/2021 5:55:37 PM

[Submissions](#)

[Details](#)

sa

Due: 7/31/2021 9:36:15 PM TotalMarks: 12

Date Created: 7/29/2021 5:35:48 PM

[Submissions](#)

[Details](#)

Renders all the assignment modules where an instructor can view all user submissions for assignments. Also provides a button to create a new assignment.

- **SingleFeedbackPage.js**

← → ⌂ ⌂ Page 1 of 2

CSCC63 ASSIGNMENT 2
DUE 11:59PM, JULY 16

Warning: Your electronic submission of a PDF to Crowdmark (through Quercus) affirms that this assignment is your own work and no one else's, and is in accordance with the University of Toronto Code of Behaviour on Academic Matters, the Code of Student Conduct, and the guidelines for avoiding plagiarism in CSC03. Note that using Google or any other online resource is considered plagiarism.

1. (20 marks) Let $L_1 = \{ (G) \mid G \text{ is a CFG such that for some } w \in \Sigma^*, \text{ both } w \text{ and } \bar{w} \text{ are in } L(G) \}$. Determine whether L_1 is recognizable, co-recognizable, both, or neither. Prove your answer.

2. (10 marks) Prove the following:

$$\overline{\text{ALL-TM}} \leqslant \{ (M) \mid |L(M)| = \infty \}.$$

3. (20 marks) Let L_3 be defined as:

$$L_3 = \{ (M) \mid L(M) \text{ is context-free} \}.$$

Determine whether L_3 is recognizable, co-recognizable, both, or neither. Prove your answer.

4. (15 marks) Let TWIN-PATH = $\{ (G = (V, E), s, t, k) \mid G \text{ is a directed graph, and there are at least two edge-disjoint paths from } s \text{ to } t \text{ of path length } k \}$.

(a) (5 marks) Show that TWIN-PATH $\in \text{NP}$.

(b) (10 marks) Assuming that TWIN-PATH is NP-COMPLETE, show that TWIN-PATH is NP-COMPLETE.

5. (15 marks) Let POWER-PATH = $\{ (G = (V, E), s, t) \mid G \text{ is a directed graph, and } \exists n \in \mathbb{N}, \text{ such that } G \text{ has an } s \text{ to } t \text{ path of length } 2^n \}$.

(a) (5 marks) Show that POWER-PATH $\in \text{NP}$.

(b) (10 marks) Assuming that TWIN-PATH is NP-COMPLETE, show that POWER-PATH is NP-COMPLETE.

6. (15 marks) Let EDGE-DENSE = $\{ (G = (V, E), m, n) \mid G \text{ is an undirected graph with an } n\text{-vertex subgraph that has at least } m \text{ edges} \}$.

(a) (5 marks) Show that EDGE-DENSE $\in \text{NP}$.

(b) (10 marks) Assuming that CLIQUE is NP-COMPLETE, show that EDGE-DENSE is NP-COMPLETE.

7. (5 marks) Consider the language

$$\text{FACT-RANGE} = \{ (n, a, b) \mid n, a, b \in \mathbb{N}, \text{ and } \exists k \in \mathbb{N}, a \leq k \leq b \text{ and } k \text{ divides } n \}.$$

Now, consider the following program to solve FACT-RANGE:

```

FIND-FACT—“On  $(n, a, b)$ :
1. If  $(b < a)$  or  $(b < a)$ :
   2. Reject.
   3. For  $i = \max(a, 1)$  to  $b$ :
      4. If  $n \% i == 0$ :
         5. Accept.
         6. Reject.

```

Is this a polytime algorithm? Why or why not?

Grade:

[Submit](#)

Title: A2
Submitted by: davidtan
Grade: 33 / 60 (55.00%)
Current Feedback:
nice try, try again next year. cya there TEST instructor only

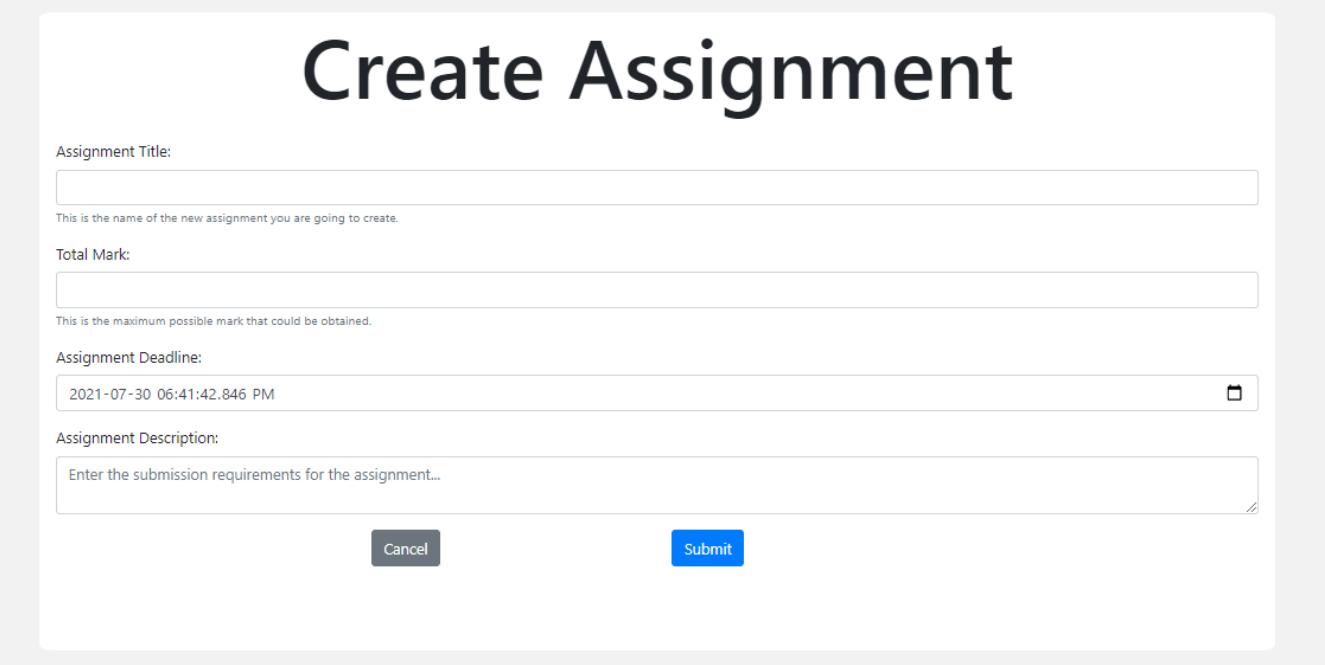
2021/07/25

Submit Feedback
Feedback:
Additional notes about the submission...
Grade:

[Submit](#)

Renders 3 components, the feedbackPdf, feedback, and feedbackForm modules. After selecting a user's submission from the DeliverableFeed, this page displays all the specific information about the user submission chosen. It also renders the feedback submission form which updates this submission's feedback and grade.

- **DlbsCreate.js**



The screenshot shows a modal window titled "Create Assignment". The form contains the following fields:

- Assignment Title:** A text input field with placeholder text: "This is the name of the new assignment you are going to create."
- Total Mark:** A text input field with placeholder text: "This is the maximum possible mark that could be obtained."
- Assignment Deadline:** A date input field showing "2021-07-30 06:41:42.846 PM" with a calendar icon to its right.
- Assignment Description:** A text area with placeholder text: "Enter the submission requirements for the assignment..."

At the bottom of the form are two buttons: "Cancel" (gray) and "Submit" (blue).

Renders a form that allows instructors to create and submit assignments.

- **AssignmentFeed.js**

Assignments

Tets assignmnet 1 Due: 7/31/2021 10:09:10 PM TotalMarks: 12	Date Created: 7/30/2021 1:55:36 PM
Submissions	Details
final assignment test Due: 7/10/2021 9:58:28 PM TotalMarks: 12	Date Created: 7/29/2021 5:55:37 PM
Submissions	Details
sa Due: 7/31/2021 9:36:15 PM TotalMarks: 12	Date Created: 7/29/2021 5:35:48 PM
Submissions	Details

Renders all the assignment modules where it shows only the user's submission to an assignment.

- **SingleUserSubmissionPage.js**

The screenshot shows a web page titled "Assignment: sa". The page header includes navigation icons (back, forward, search) and a page number "Page 1 of 5". The main content area displays the assignment details:

- Title:** sa
- Submitted by:** christophersuh
- Grade:** 12 / 12 (100.00%)
- Current Feedback:** Good job on your assignment!

Below the title, there is a detailed assignment description:

CSCC42 UTSC Summer 2021
Assignment 3
Database Design Theory and Normalization

Assignment Details:

- Due Date:** August 2 11:59pm
- Points:** 100
- Read the note on academic integrity: <https://utsc.utoronto.ca/aacc/academic-integrity-matters>
- Instructions:** Late submissions will be penalized 10% for each day up to 3 days. Any submissions after **5 August** will not be marked.
- There are three parts in this assignment.
 - Part 1 requires you to design ER Diagram using any online drawing tool.
Note: Handwritten solutions/diagrams will not be accepted.
 - Part 2 requires conversion from ER diagram to relational schema and improved version of ER diagram.
There are 3 questions that need to be answered including analysis of redundancies.
 - Part 3 requires you to answer questions on functional dependencies and normal forms.
There are 3 questions that need to be answered with subparts.

Other instructions include:

- You can work in a group of up to 3. Please submit only one copy of your solutions per team.
- Submission:** <https://imarkus.utm.utoronto.ca/csc43921/>
- For this assignment, you need to submit all typed solutions in a single PDF named **Assignment3.pdf**. You can use LaTeX or any other word processing software. ER Diagrams should be computer generated as well and should not be hand drawn.

Renders two components. It utilizes the feedback and feedbackPdf modules to display information. After a user selects their own user submission from the AssignmentFeed, this page displays all the specific information of their user

submission chosen.

- **DlbsPage.js**

final assignment test

final assignment test

Instructor: christophersuh

Total Marks: 12

sadsd

Assignment Deadline: 7/10/2021 9:58:28 PM

Submit your work:

Select the file to upload:

No file chosen

Renders the deliverable details page where it contains the Dlbs module where it gets the information from the assignment that the user selected to get to this page. It also allows you to submit a pdf of your work to make user submission.

- **MessagePage.js**

Messages

Recent Messages

User: abc

Name: Emily Gilbert

User: ddd

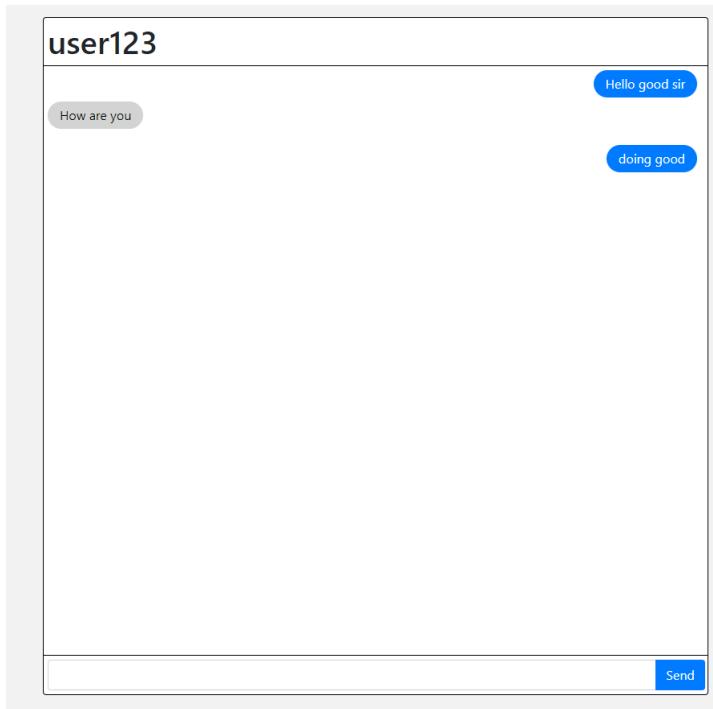
Name: Jennifer Blodgett

User: user123

Name: John Doe

This component renders the user's recent messages from other people. The New message button will redirect to the user search page to find new people to message.

- **MessageSection.js**



This component renders the messages sent and received to another user. The user can send messages to another person through the input box.

2. Sub-Components

- **Groups.js**

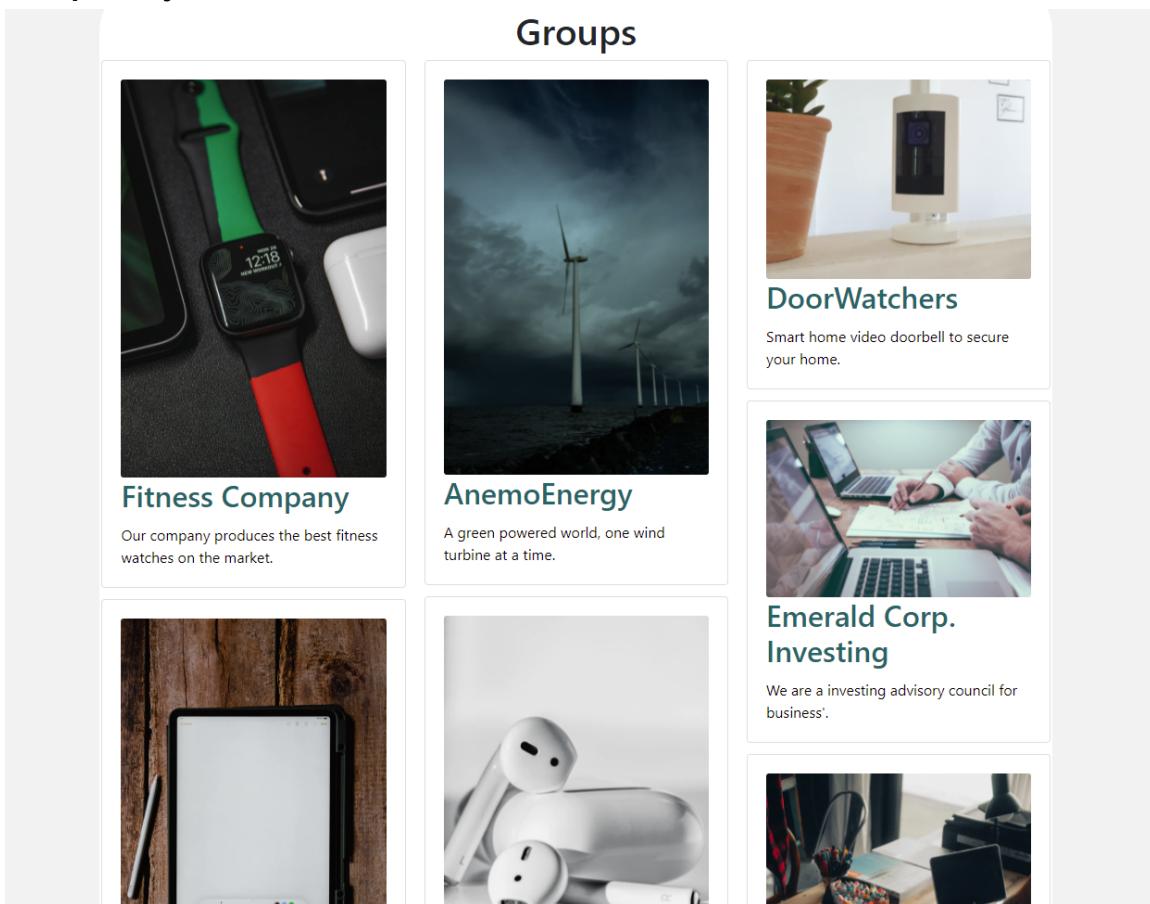


DrawIO

A mobile drawing app that allows users to draw using AI based assistance.

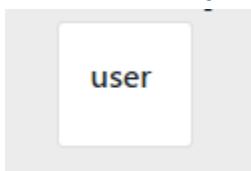
This component shows the group name, group photo and About section

- **GroupsList.js**



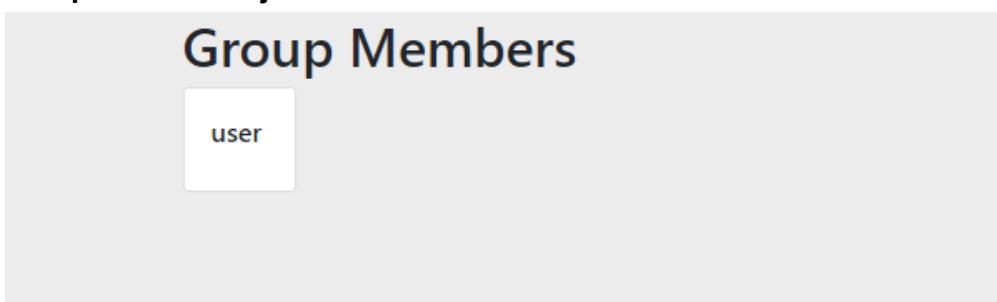
Renders a list of groups from the database.

- **GroupMember.js**



Renders the user's name in a small compact way.

- **GroupMemberList.js**



Renders list of all group members passed into it. Note: `members = [{member1, ..., member}]`

- **Group.js**



DoorWatchers

MemberID:

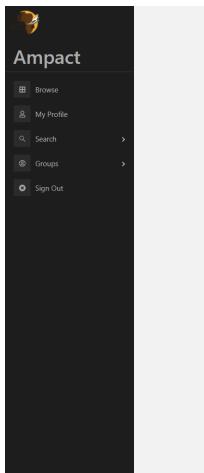
Add member Remove Member Edit

About

Smart home video doorbell to secure your home.

The Group.js component takes in groupName, about, picture, members = [{member1, ..., member}] params. It then displays this information using html/React code.

- **DefaultPage.js**



Helps render the layout for the navigation bar.

- **UserPage.js**

A screenshot of a user profile page. At the top is a large banner image of a sunset over water. Below the banner is a circular profile picture of a person's face. To the right of the profile picture, the name "John Doe" is displayed, followed by the role "entrepreneur" and a blue "Edit Profile" button. Below this section is a white box containing the heading "About Me" and the text "Hi it's me!".

Actually renders the module for UserProfile.js. It displays the user's First name, last name (or username if it isn't changed), the user role, and about me, a profile picture and a default banner photo.

- **VideoPreview.js**



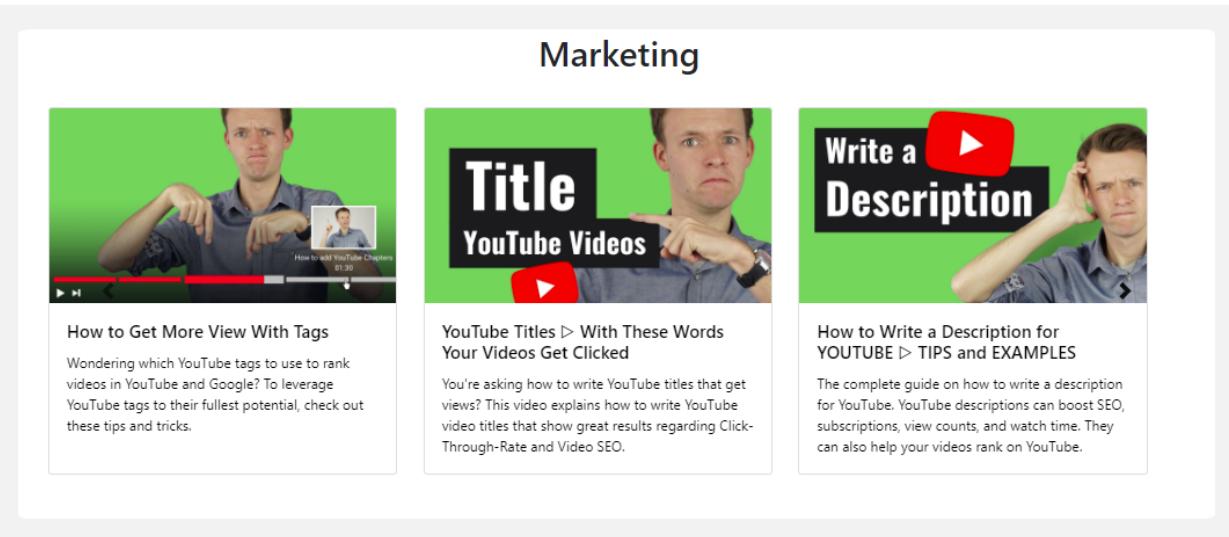
Title
YouTube Videos

YouTube Titles ▷ With These Words
Your Videos Get Clicked

You're asking how to write YouTube titles that get views? This video explains how to write YouTube video titles that show great results regarding Click-Through-Rate and Video SEO.

Renders a video module for Browse.js which contains a video thumbnail, the video title and its description.

- **VideoTagSection.js**



Marketing



How to Get More View With Tags
Wondering which YouTube tags to use to rank videos in YouTube and Google? To leverage YouTube tags to their fullest potential, check out these tips and tricks.



**YouTube Titles ▷ With These Words
Your Videos Get Clicked**
You're asking how to write YouTube titles that get views? This video explains how to write YouTube video titles that show great results regarding Click-Through-Rate and Video SEO.



**Write a
Description**
How to Write a Description for YOUTUBE ▷ TIPS and EXAMPLES
The complete guide on how to write a description for YouTube. YouTube descriptions can boost SEO, subscriptions, view counts, and watch time. They can also help your videos rank on YouTube.

Renders the video carousel module for Browse.js which you can select and scroll left or right using the arrows through videos related to a certain tag. It also automatically scrolls as time passes.

- **UserProfileList.js**

Lara Kristine Gomez Lara Kristine Gomez	David Russ David Russ	Brian Williams Brian Williams
Christopher Suh Christopher Suh	Timothy Leja Timothy Leja	Jessie Parsons Jessie Parsons
David Tan David Tan	Carl Morris Carl Morris	Charles Driscoll Charles Driscoll
Lawrence Cai Lawrence Cai	Mary Stambaugh Mary Stambaugh	Rose Connor Rose Connor
Raymond Ma Raymond Ma	Paul Heller Paul Heller	Jerry Owens Jerry Owens
Selina Zhi Selina Zhi	David Kelly David Kelly	James Berk James Berk
Jane Armstrong Jane Armstrong	Leroy Breceda Leroy Breceda	Mark Kirk Mark Kirk

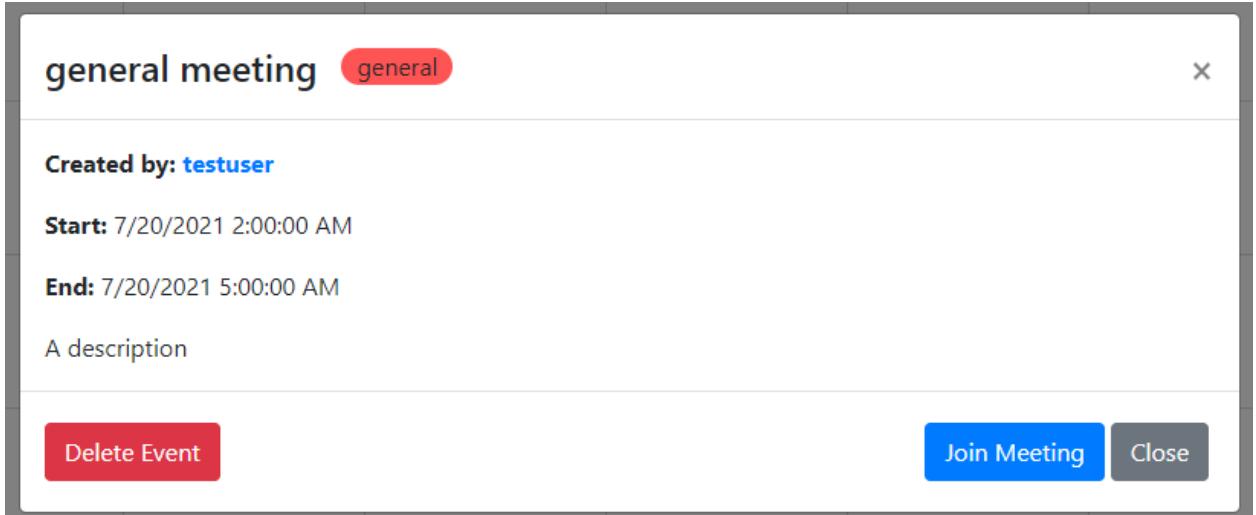
This creates the module that renders the 3 column list of profiles when searching for user profiles.

- **UserProfileItem.js**

David Tan David Tan

This creates the individual user profiles in the user profile list. It displays their first name and last name

- **EventPopup**



Shows a calendar's event detail in a modal window popup. Triggers when the user clicks on an event in the calendar. Shows start time, end time, description, creator, and meeting link if there is any.

- **AddEventPopup**

A screenshot of a modal window titled "Add Event". It contains fields for Event Title (with placeholder "Title of this new event"), Event Type (radio buttons for general, personal, group), Start Date and End Date (date pickers), Event description (text area with placeholder "Additional Notes about the event..."), and Conference Link (text input field with placeholder "Input conference link"). At the bottom are "Update" and "Close" buttons.

Shows options to create a new calendar event in a modal window popup. Triggers when the user clicks on the Add Event button on the calendar page. Provides access to specify event title, event type, start date, end date, description, creator, and meeting link if there is any.

- **Post.js**

A screenshot of a post card component. At the top left is a circular profile picture of a person with blonde hair. To its right is the username "davidtan". Below the profile picture is the text "Type: announcement". In the top right corner are the date "2021/07/15" and the visibility setting "Visible to: all". The main message content is "hello". At the bottom left of the card is a blue rectangular button labeled "Comments".

Renders the information for an individual post: visibility, date, username of poster, type, message. It also allows for the expansion and collapse of the comment section for that post.

- **CommentSection.js**

A screenshot of a comment section component. At the top left is a blue rectangular button labeled "Comments". Below it is a form with a text input field containing "Write your comment here!" and a blue "Submit" button to its right. A horizontal line separates this from the comments below. The first comment is by user "christophersuh", shown with a circular profile picture, the name "christophersuh", the date "2021/07/15", and the message "hello". Below this comment is another blue rectangular button labeled "Replies". The second comment is also by "christophersuh", with the same profile picture, name, date, and message "hello 2". Below this second comment is another blue rectangular button labeled "Replies".

Renders the information for the comment section with all the comments for that specific post. The comment section is expandable and collapsible and it also provides the ability to create new comments for that post.

- **ReplySection.js**

The screenshot shows a user interface for managing comments. At the top, there's a comment by user 'christophersuh' posted on '2021/07/15' with the message 'hello'. Below this, a blue button labeled 'Replies' is visible. A reply section is expanded, showing three replies from the same user 'christophersuh' on '2021/07/15': 'hii', 'sd', and 'sd'. Each reply includes a small profile picture of the user. To the right of each reply, the date '2021/07/15' is displayed. At the bottom of the reply section, there's a text input field with the placeholder 'Write your reply here!' and a blue 'Submit' button.

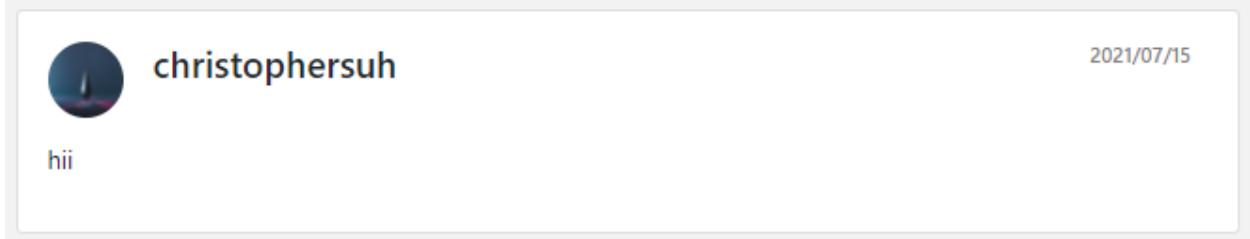
Renders the information for the replies section with all the replies for that specific comment. The replies section is expandable and collapsible and it also provides the ability to create new replies for that comment.

- **Comment.js**

The screenshot shows a user interface for managing comments. At the top, there's a comment by user 'christophersuh' posted on '2021/07/15' with the message 'hello'. Below this, a blue button labeled 'Replies' is visible. A replies section is collapsed, indicated by a small arrow icon next to the button. The rest of the screen is blank, suggesting the user has collapsed the replies section.

Renders the information for an individual comment: date, username of poster, message. It also allows for the expansion and collapse of the replies section for that comment.

- **Reply.js**



Renders the information for an individual reply: date, username of poster, message.

- **ChatBubble.js**

How are you doing good

Renders each individual message. There are two styles of chat bubbles: send (left) and received (right).

- **Deliverable.js**

Sample task 3	Date Created: 7/27/2021 7:48:20 PM
Due: 8/3/2021 12:00:00 AM TotalMarks: 20	
Submissions	Details

Renders the information for a single deliverable: title, dueDate, totalMarks, date created.

- **SubmissionSection.js**

Sample task 3

Date Created: 7/27/2021 7:48:20 PM

Due: 8/3/2021 12:00:00 AM TotalMarks: 20

Submissions

Details

Search for user submission

Search



222

Submission date: 7/28/2021 8:08:24 PM

Grade: 18 / 20



davidtan

Submission date: 7/28/2021 7:56:02 PM

Grade: / 20



111

Submission date: 7/29/2021 3:05:04 PM

Grade: / 20

Renders all of the user submissions for a specific assignment. Also has a search bar to search for a specific user submission.

- **Submission.js**



[222](#)

[Submission date: 7/28/2021 8:08:24 PM](#)

[Grade: 18 / 20](#)

Renders data for a single submission preview. Includes; username, submission date, grade, totalMarks

- **UserDeliverable.js**

Merge Assignment

Date Created: 7/29/2021 4:21:23 PM

Due: 7/29/2021 8:00:00 PM TotalMarks: 100

Submissions

Details

Renders the information for a single deliverable: title, dueDate, totalMarks, date created. Except it only has submissions pertaining to a single user.

- **UserSubmission.js**

Merge Assignment Date Created: 7/29/2021 4:21:23 PM

Due: 7/29/2021 8:00:00 PM TotalMarks: 100

Submissions	Details
 christophersuh	Submission date: 7/29/2021 5:49:53 PM Grade: 46 / 100

Renders submissions for a specific assignment and user.

- **feedbackForm.js**

Submit Feedback

Feedback:

Additional notes about the submission...

Grade:

Submit

Renders the feedback form utilized in SingleFeedbackpage. This allows for users to update a user's submission's grade and provide them with text feedback about their submission. The grade that they submit must not be greater than the total grade that the assignment is worth. They must also have a grade submission in order to submit the form

- **feedback.js**

Title: A2

2021/07/25

Submitted by: davidtan

Grade: 33 / 60 (55.00%)

Currrent Feedback:

nice try, try again next year. cya there TEST instructor only

Renders all the information about a user's submission. It provides the title of the assignment and date for the assignment that a user is submitting to. It also tells you which user made this submission, the grade that it has and the current feedback.

- **feedbackPdf.js**

CSCC63 ASSIGNMENT 2

DUE 11:59PM, JULY 16

Warning: Your electronic submission of a PDF to Crowdmark (through Quercus) affirms that this assignment is your own work and no one else's, and is in accordance with the University of Toronto Code of Behaviour on Academic Matters, the Code of Student Conduct, and the guidelines for avoiding plagiarism in CSCC63. Note that using Google or any other online resource is considered plagiarism.

1. (20 marks) Let $L_1 = \{\langle G \rangle \mid G \text{ is a CFG such that for some } w \in \Sigma^*, \text{ both } w \text{ and } 0w \text{ are in } L(G)\}\}.$

Determine whether L_1 is recognizable, co-recognizable, both, or neither. Prove your answer.

2. (10 marks) Prove the following:

$$\overline{\text{ALL}_{\text{TM}}} \leqslant_m \{\langle M \rangle \mid |\overline{L(M)}| = \infty\}.$$

3. (20 marks) Let L_3 be defined as:

$$L_3 = \{\langle M \rangle \mid L(M) \text{ is context-free}\}.$$

Determine whether L_3 is recognizable, co-recognizable, both, or neither. Prove your answer.

4. (15 marks) Let TWIN-PATH = $\left\{ \langle G = (V, E), s, t, k \rangle \mid \begin{array}{l} G \text{ is a directed graph, and there are at least two} \\ \text{edge-disjoint paths from } s \text{ to } t \text{ path of length } k \end{array} \right\}.$

- (a) (5 marks) Show that TWIN-PATH $\in \text{NP}$.

- (b) (10 marks) Assuming that HAM-PATH is NP-COMPLETE, show that TWIN-PATH is NP-COMPLETE.

5. (15 marks) Let POWER-PATH = $\left\{ \langle G = (V, E), s, t \rangle \mid \begin{array}{l} G \text{ is a directed graph, and } \exists n \in \mathbb{N}, \\ G \text{ has an } s \text{ to } t \text{ path of length } 2^n \end{array} \right\}.$

- (a) (5 marks) Show that POWER-PATH $\in \text{NP}$.

- (b) (10 marks) Assuming that HAM-PATH is NP-COMPLETE, show that POWER-PATH is NP-COMPLETE.

6. (15 marks) Let EDGE-DENSE = $\left\{ \langle G = (V, E), m, n \rangle \mid \begin{array}{l} G \text{ is an undirected graph with an } n\text{-vertex} \\ \text{subgraph that has at least } m \text{ edges} \end{array} \right\}.$

- (a) (5 marks) Show that EDGE-DENSE $\in \text{NP}$.

- (b) (10 marks) Assuming that CLIQUE is NP-COMPLETE, show that EDGE-DENSE is NP-COMPLETE.

7. (5 marks) Consider the language

$$\text{FACT-RANGE} = \{\langle n, a, b \rangle \mid n, a, b \in \mathbb{N}, \text{ and } \exists k \in \mathbb{N}, a \leq k \leq b \text{ and } k \text{ divides } n\}.$$

Now, consider the following program to solve FACT-RANGE:

```
FIND-FACT= "On ⟨n, a, b⟩:
    1. If (0 < n < a) or (b < a):
        2.   Reject.
    3. For i = max(a, 1) to b:
        4.   If n % i == 0:
            5.       Accept.
        6.   Reject."
```

Is this a polytime algorithm? Why or why not?

Renders the pdf of the user's submission. It provides a navigation tab at the top where you can select different pages and also zoom in and out of the pdf.

- **Dlbs.js**

Test video 2

Test video 2

Instructor: christophersuh

Total Marks: 12

saa

Assignment Deadline: 7/30/2021 9:33:27 PM

Submit your work:

Select the file to upload:

AfricanImpactChallenge.pdf

Filetype: application/pdf

Size in bytes: 163516

Last Modified Date: 5/21/2021

This component helps render all the information and modules for the DLbsPage. It provides all the assignment detail information from title, to the instructor that created the submission, total marks, the assignment description and deadline. It allows for a submission form where you can choose and upload a pdf file where it also provides info about the pdf you are about to submit.