ProudSmart

Learning Platform

Team JABB
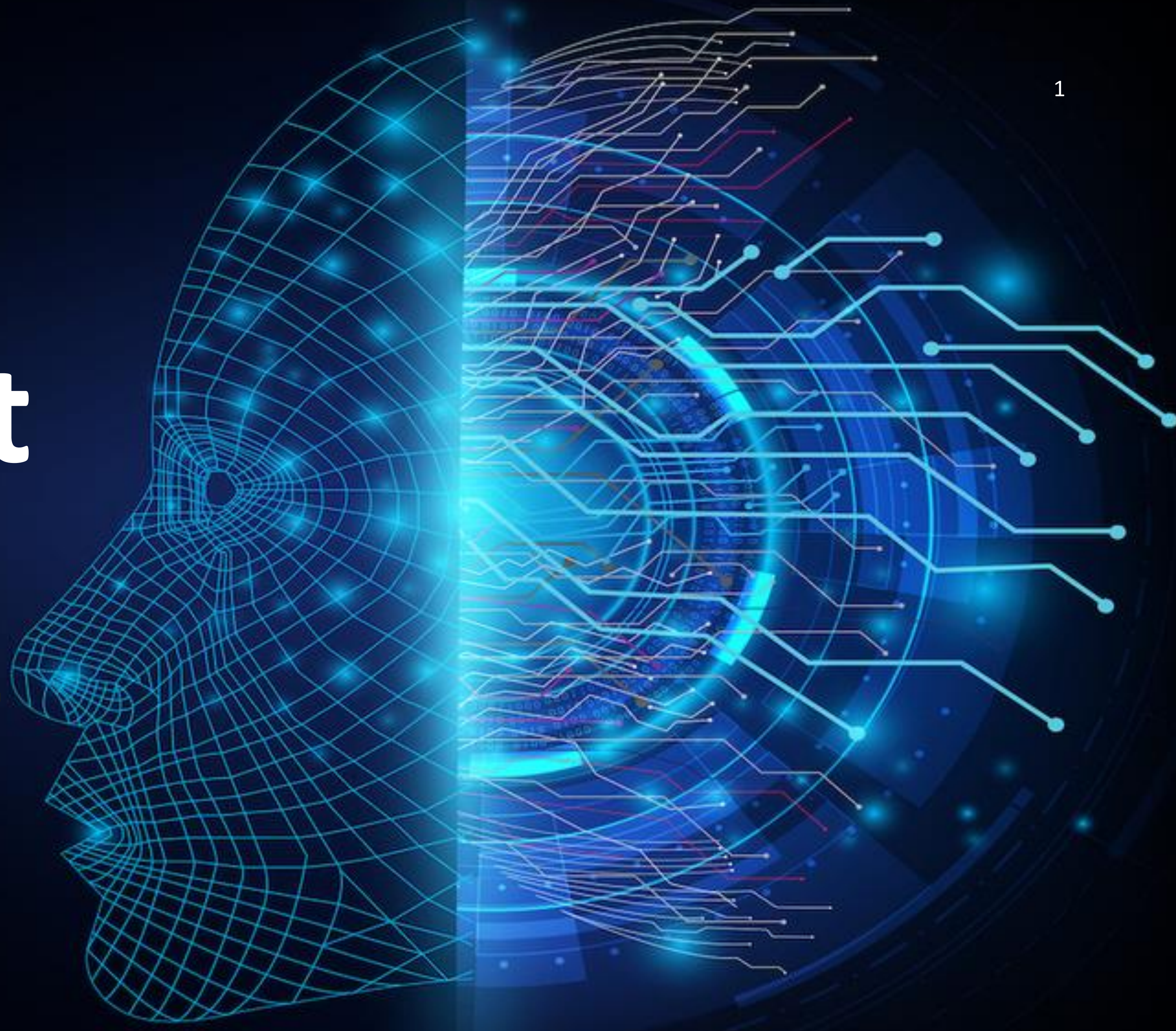
Table of Contents

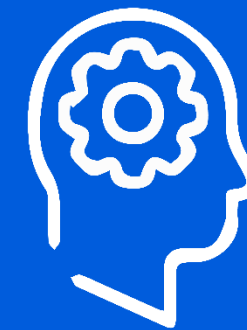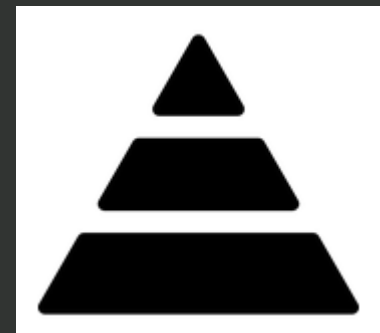# The Client Problem

ProudSmart is a business which sells complex technology solutions

A online learning platform will enable ProudSmart to drive more value out propriety technologies through educating:
- Internal staff
- Customers
- Partners
- Students

Project Goals

Our MVP focused on 4 critical features…..

1 **Permission based access**

Different user types have different access to course content

2 **Live streaming of course content**

Leveraging third party services to host and stream content

3 **Payments**

Ability to accept payments from users

4 **Content creation**

Ability for educators to upload course content

Design Considerations

Our design focus was to provide a simple and intuitive front-end user experience similar to other popular online learning portals (e.g. Udemy, Coursera, Edx etc…)
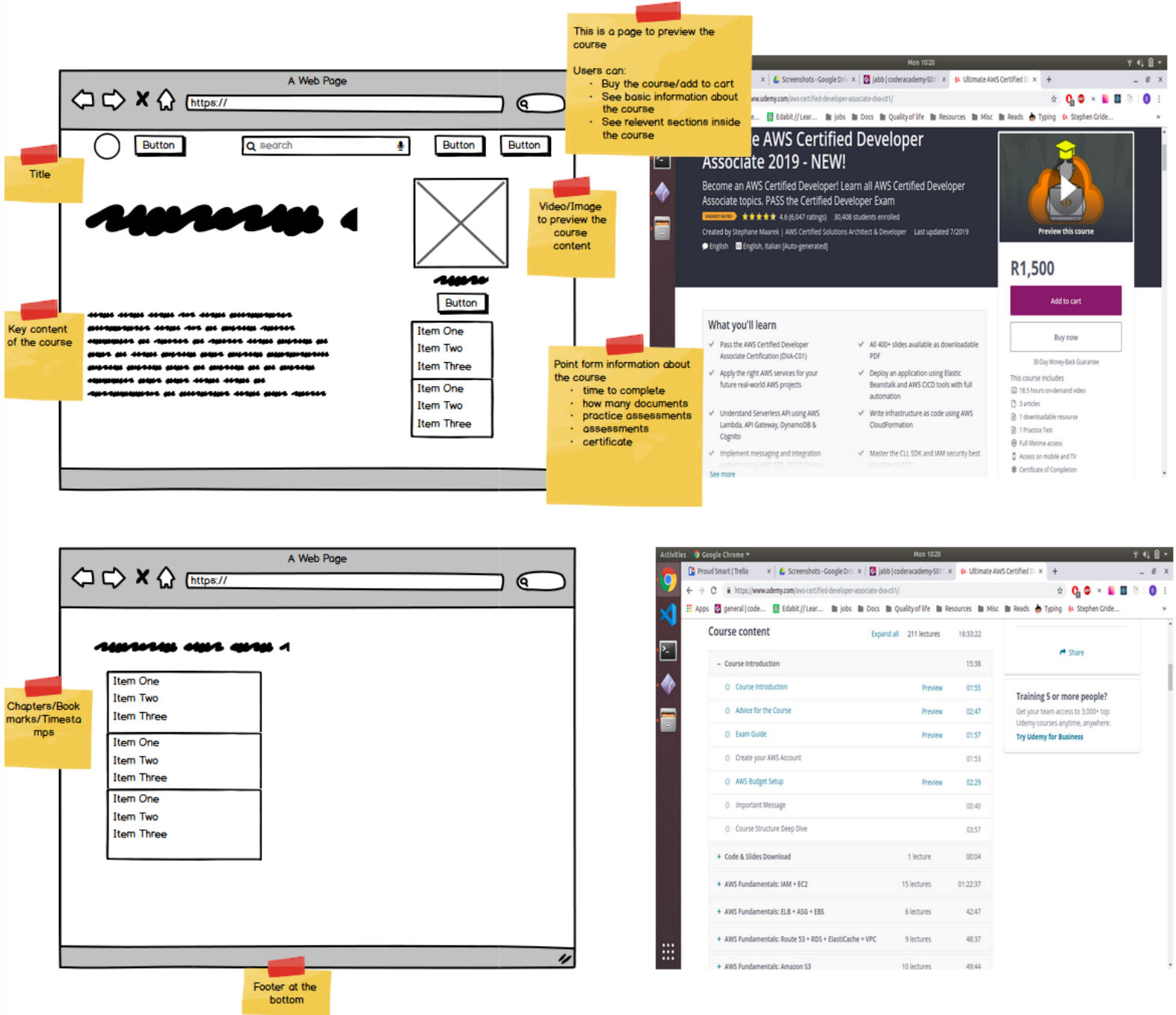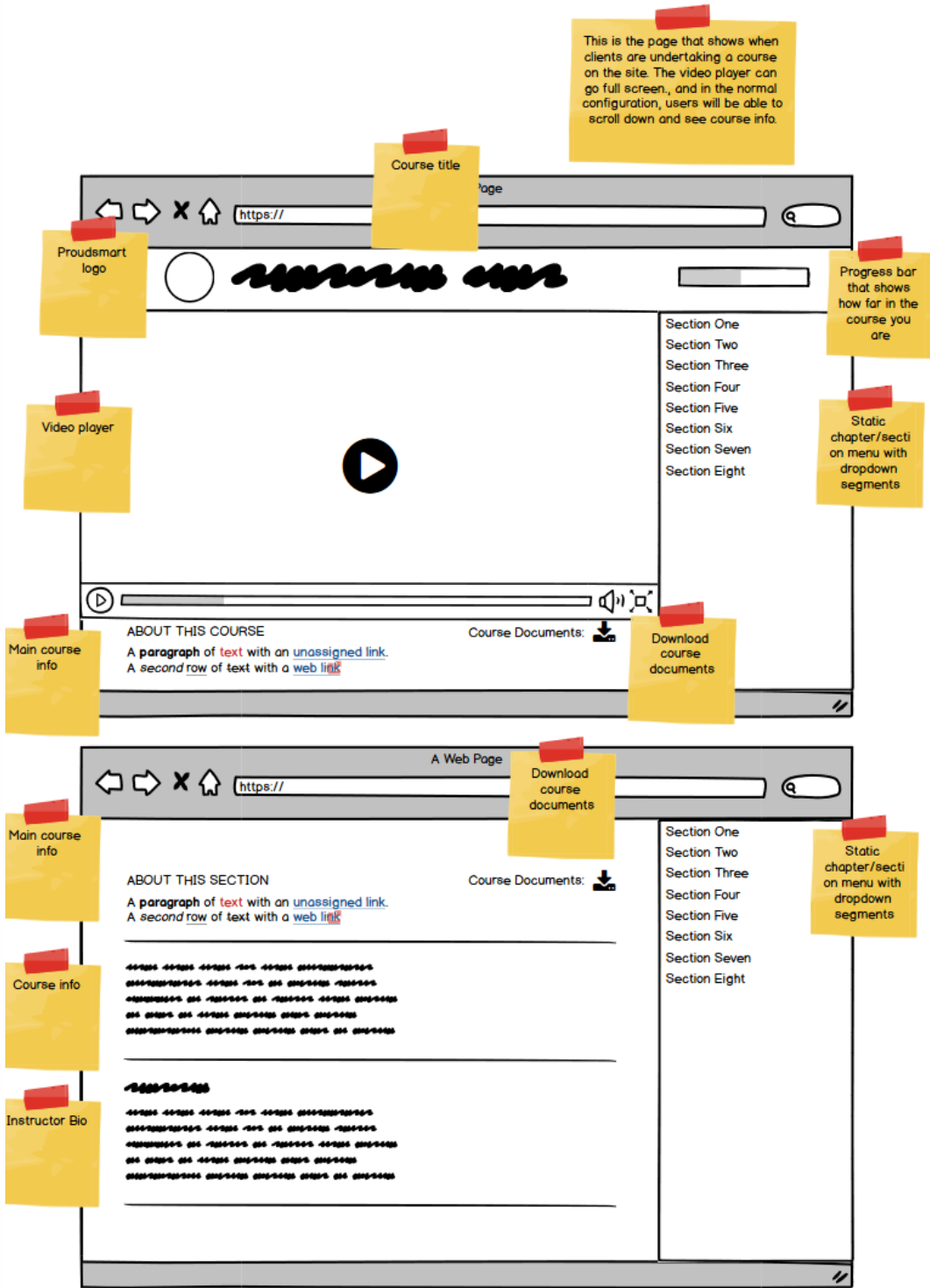
Project Workflow

We separated front and back end responsibilities to simplify project delivery and reduce overall rework

Josh

Ari   **B**ruce

Billy

Third Party Services

Front End Experience

API Service
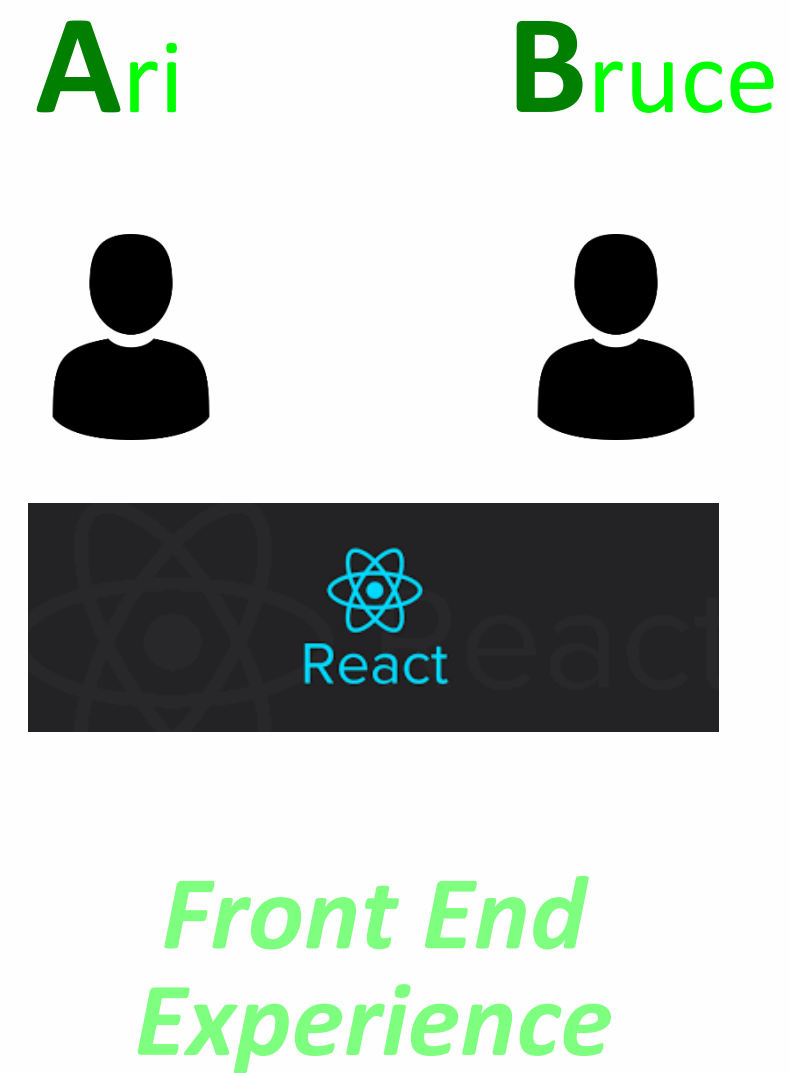
Key Challenges

- Implementing live streaming

- Handling state

- Addressing roadblocks

- Managing client expectations

Ethical Issues

- Accessibility considerations

- Piracy risk

- Content censorship

- Disguised sales medium

Favorite Parts

Investing lot's of time into planning early on in the project ensured strong teamwork and communication throughout the project

	A	B	C	D	E	F	G	H

							USER ACCESSS				
Method	Base	Route Part 1	Route Part 2	:id?	Page	Features	Form Data	Non User	Free	Paid	Teacher	Admin
POST	/	auth	register	n	Register	> Enables a user to sign up > Option to sign up using oAuth	> Email address (string) > First name (string) > Last name (string) > Password (string) > Profile photo (file) > Interest tags (dropdown - array of strings)	y	n	n	n	n
POST	/	auth	educator_application	n	Educator Application	> Enables user to apply to be an educator	> Qualifications (array of strings) > About me (text) > Capable tags (array of strings)	n	y	y	y	y
PUT	/	auth	edit	y	-	> Route to update account info	> All of register form data > All of educator application data	n	y	y	y	y
GET	/	admin	dashboard	n	Admin Dashboard	> Buttons to index pages: -> Courses -> Teachers -> Modules -> Users > Notifications -> Pending application	-	n	n	n	n	y
GET	/	users	-	n	Users Index	> List of all users > For each user there will be a button: -> Edit -> Contact -> Delete	-	n	n	n	n	y
GET	/	users	dashboard	n	User Dashboard	> List of courses you signed up for > Recommened course (based on interested mod > Button to apply to be a teacher	-	n	y	y	n	n
GET	/	users	edit-account-info	y	User Edit	> Edit account info > Edit teacher profile (if approved teacher) > GET request returns the user's account info to p	> All of register form data > All of educator application data	n	y	y	y	y

Live Walkthrough

Question Time