

Fall 2024 CastNXT Final Report

Rahul Baid, Jose Salazar, Shweta Kumaran, Harsh Shah, Alea Nablan, Rituparna Mandal, Milad Saadat and Fatemah Zare

Summary:

CastNXT is an all-in-one web application and platform for event plan seamlessly. The application serves to automate the process of connecting producers, clients, and talents to plan and execute these events. The application meets these connections by enabling producers to create events, synchronize with the clients to shortlist talents, and remunerate them. Clients are also able to select talents based on their preferences. Additionally, talents can search and sign up for events.

The client of the project is Tito Chowdhury, the owner of FashionNXT. Our client is our key stakeholder in this project. However, the talents, clients, and producers utilizing this application are also stakeholders. The application has currently served the communication and event planning portion. This semester, the client and stakeholder wanted the application to be production-ready, streamline the communication between producers, clients, and talents, and implement a few features to enhance the user experience.

All user stories:

Sprint 1:

1. Deploy the app to Heroku

As a developer,
I want to deploy the application to Heroku,
so that it is accessible to users via the web and I can ensure the app works as expected in the production environment.

Points: 3

Assigned To: Jose Salazar

Status: Complete

2. Deploy the app to the Local System.

As a developer,
I want to deploy the application to the Local System,
So that it is accessible to users and I can ensure the app works as expected in the production environment in the Local System

Points: 4

Assigned To: All the Team Member
Status: Complete

3. Update the library versions

As a user,
I want to secure my data by keeping libraries up-to-date
So that it is not exploited by malicious actors.

Points: 2
Assigned To: Rahul Baid
Status: Complete

4. Connect MongoDB to the App

As a user,
I want the app to be connected to a database
So that I do not have to face any internal issues when trying to register or log in to the system.

Points: 2
Assigned To: Jose Salazar
Status: Complete

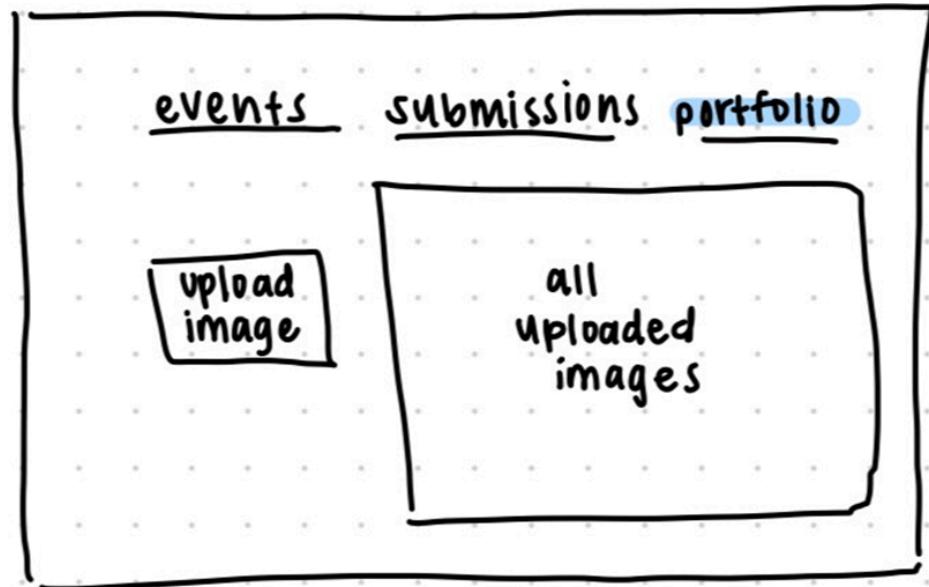
5. Update Feature and Code Documentation

As a user,
I want to access up-to-date documentation about the functionalities,
so that all stakeholders (clients and admins) get the correct feature flow.
Points: 3
Status: Incomplete

6. Upload images to a User

As a user,
I want to upload images to my user page from my mobile device,
so that it is possible to add more images.
Points: 3
Status: Incomplete

Lo-fi UI mockups/storyboards:



Sprint 2:

7. Send payments

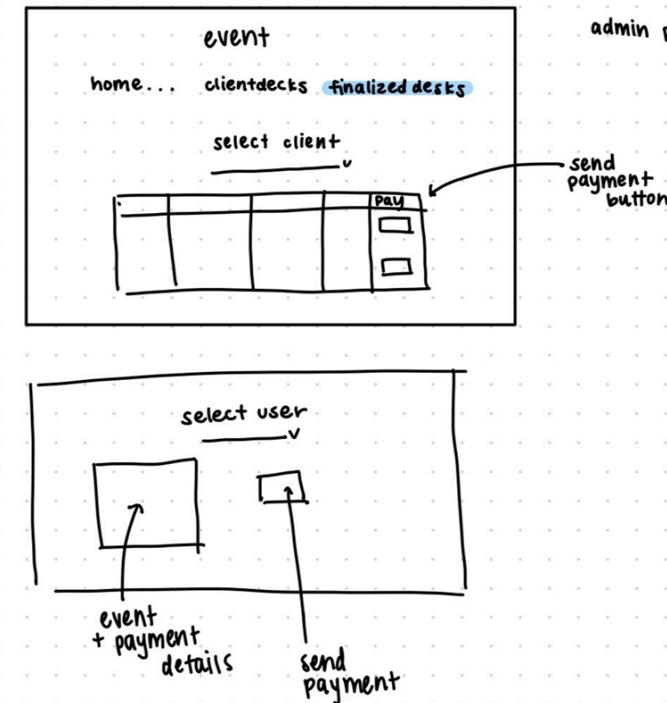
As an admin,
I want to send payment through the portal,
so that it is easily viewable by the user.

Points: 3

Assigned To: Shweta Kumaran

Status: Complete

Lo-fi UI mockups/storyboards:



Actual Screenshot:

Event 1

event 1 description
Location : Ames, Iowa
Date : 2024-10-31T05:00:00.000Z
Category : Performing Arts

[BACK TO FORMS](#)

HOME SUBMITTED DOCS SELECTED DOCS CLIENT DECKS FINALIZED DECKS SUMMARY EDIT FORM

Here you will find a report of mapped clients and talents for a certain event.

Event Summary				
Clients assigned	Payment Completed	Name	Gender	Height
client1	Mark as Completed	user123	Female	5' 2"
client1	Mark as Completed	user234	Male	5' 10"
client1	Completed	user345	Female	6' 0"

1-3 of 3 < >

Birth Date	Email	State	City	Payment
2001-10-26	user123@email.c...	Texas	Houston	<button>PAY</button>
2000-05-03	user234@email.c...	Kentucky	Louisville/Jefferso...	<button>PAY</button>
2005-05-03	user345@email.c...	Louisiana	Monroe	<button>PAY</button>

8. AI Form Generator

(The story was split into 3 i.e. Research, Proof of Concept, and development.)

Research on AI Form Generator:

As a developer,

I want to research an AI form generator that creates customizable templates based on event-specific prompts.

So that it provides insights into simplifying form creation for efficient event management.

Points: 1

Assigned To: Harsh Shah, Milad Saadat

Status: Complete

Proof of Concept on AI Form Generator:

As a developer,

I want to create a proof of concept for an AI form generator that produces customizable templates based on event-specific prompts.

So that it streamlines the form creation process, saving time and effort.

Points: 2

Assigned To: Harsh Shah, Milad Saadat

Status: Complete

Actual Code:

```
import os
import google.generativeai as genai
import re
import ast

# Set the environment variable in a Colab cell
OpenAI_API_KEY = "" # Enter OpenAPI Key

# Retrieve the API key from the environment variable
if not OpenAI_API_KEY:
    raise ValueError("API key not found. Please set the OPENAI_API_KEY environment variable.")

genai.configure(api_key=OpenAI_API_KEY)

def generate_form_fields(prompt):
    model = genai.GenerativeModel('gpt-4o-mini')
    response = model.generate_content(f"""
        Based on the following event description, generate a list of necessary fields for a Google Form.
        For each field, specify:
        1. Field name
        2. Field type (e.g., short answer, paragraph, multiple choice, dropdown, checkbox, date, time)
        3. Whether it's required or optional
        4. Any additional options or details (e.g., choices for multiple choice questions)
    """)

    return response
```

```
Provide the output as a Python list of dictionaries, where each dictionary represents a form field.
Example format:
[{"name": "Full Name", "type": "SHORT_ANSWER", "required": True}, {"name": "Email", "type": "SHORT_ANSWER", "required": True}, {"name": "Number of Guests", "type": "DROPODOWN", "required": True, "options": [1, 2, 3, 4, 5+]}, {"name": "Event Date", "type": "DATE", "required": True}, {"name": "Event Time", "type": "TIME", "required": True}, {"name": "Country", "type": "DROPODOWN", "required": True}, {"name": "State", "type": "DROPODOWN", "required": True}]

# Extract the list from the response using regex
match = re.search(r'\[\.*?\]', response.text, re.DOTALL)
if match:
    list_str = match.group(0)
    try:
        return ast.literal_eval(list_str)
    except:
        print("Error parsing the response. Here's the raw output:")
        print(response.text)
        return []
```

```
else:
    print("Couldn't find a valid list in the response. Here's the raw output:")
    print(response.text)
    return []

prompt = "Generate Google Form fields for a grand party event at country India state."
fields = generate_form_fields(prompt)

print("Generated Google Form fields:")
for field in fields:
    print(f"\nField Name: {field['name']}")
    print(f"Field Type: {field['type']}")
    print(f"Required: {field['required']}")
    if 'options' in field:
        print(f"Options: {field['options']}")
```

Development of AI Form Generator:

As a producer(admin),

I want an AI form generator that creates a customizable template based on prompts about the event.

So that it simplifies the process of getting event-specific forms, making my job easier and faster.

Points: 3

Status: Incomplete

Lo-fi UI mockups/storyboards:

Create New Event

Back to homepage AI form Generator

Step 1

Event title
Event description
Choose Date
Event State
⋮

Step 2

Choose Existing form Create New form

Add field to the following form.

Name ↑ ↓
Gender ↑ ↓

Opening new Webpage for AI form Generator.

New Web page for AI form Generator

Prompt Box

Sample Table Generator

First Name	Last Name	Contact No.	---

Regenerate again. Edit option. Save option. Download option.

9. DM Feature for Admin Communication with Clients and Talents

As a producer(admin),

I want to send messages or announcements directly to clients and talents

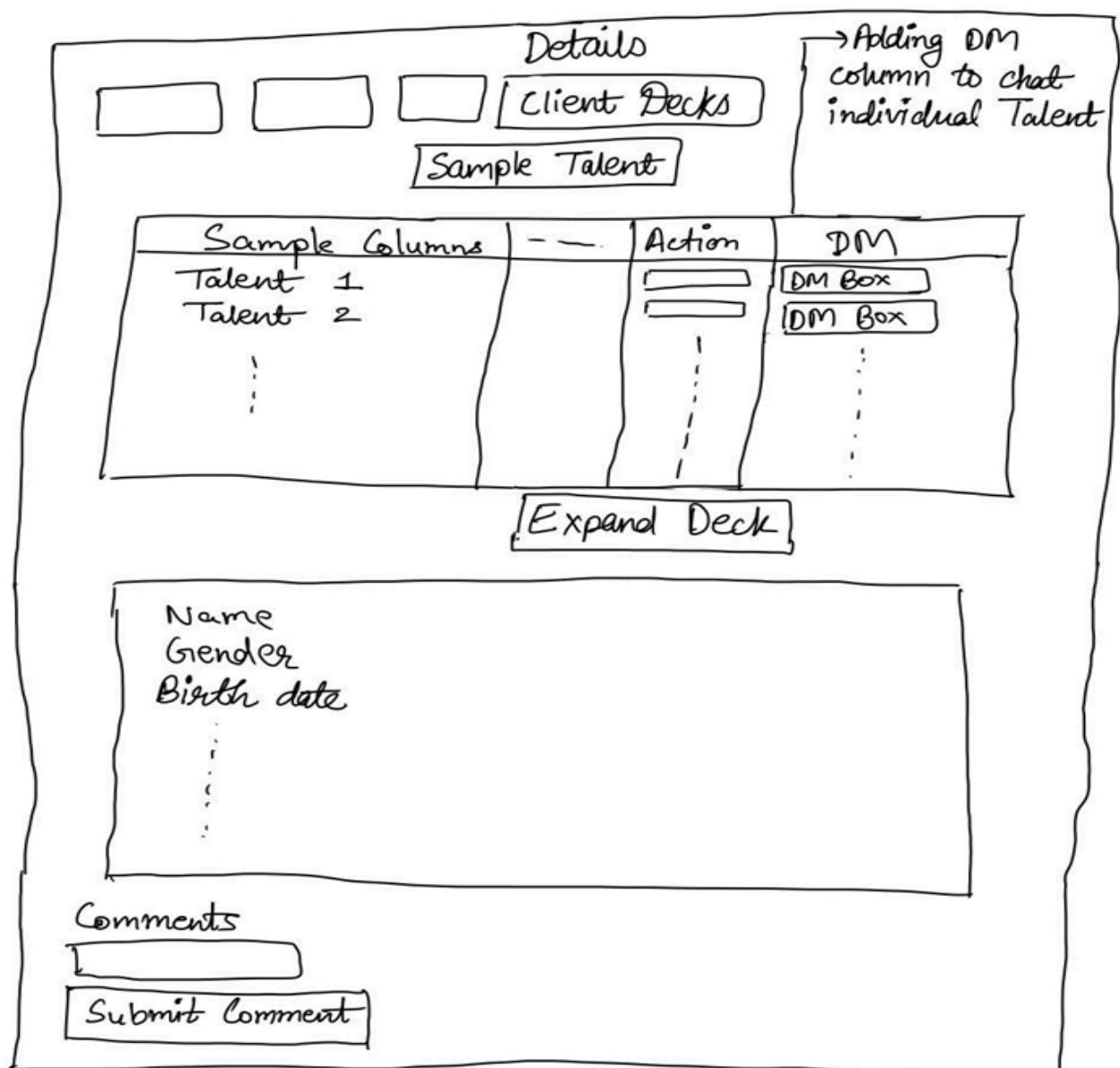
So that I can communicate important updates or information quickly and efficiently

Points: 3

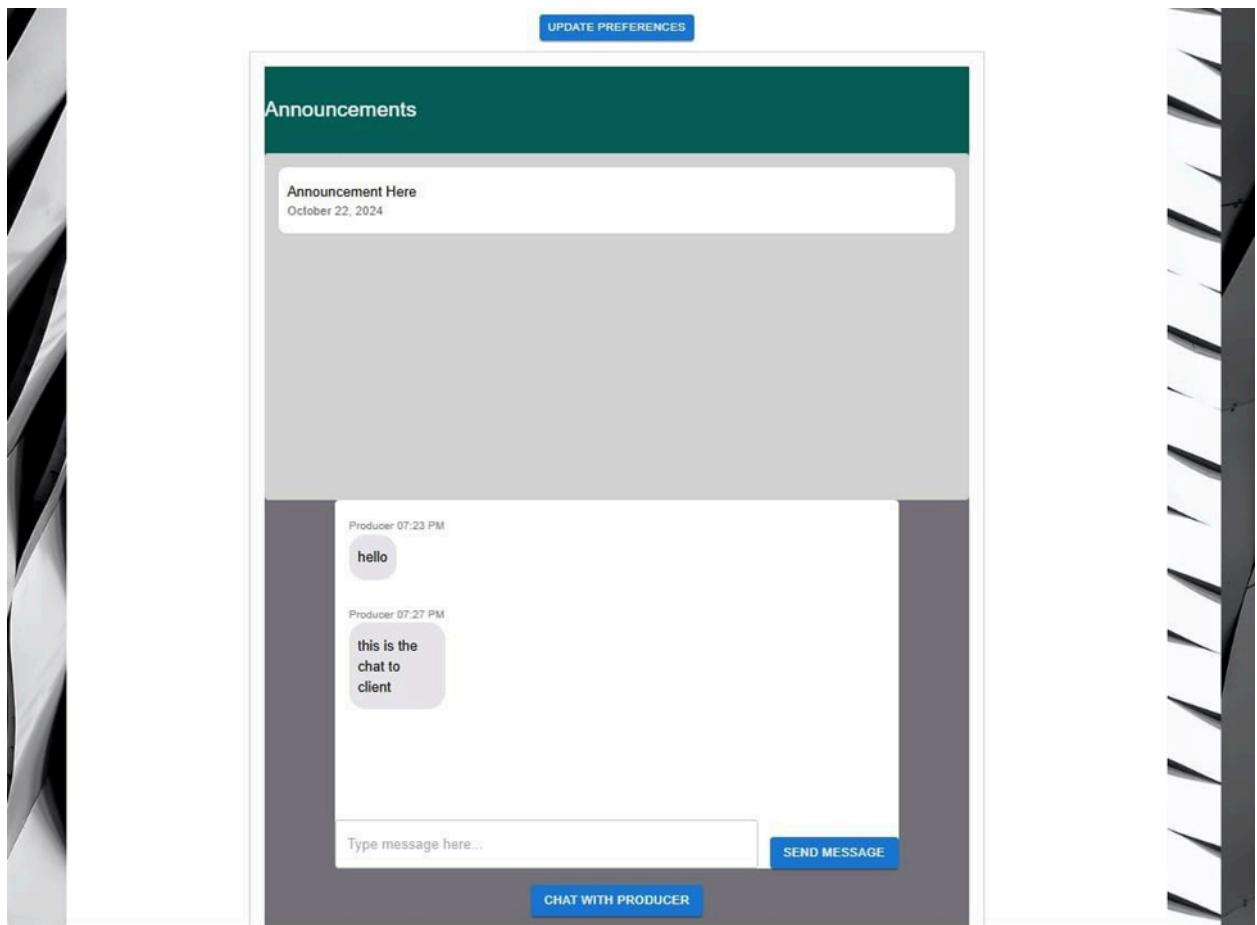
Assigned To: Jose Salazar, Rituparna Mandal

Status: Complete

Lo-fi UI mockups/storyboards:



Actual Screenshot:



10. Add Cover to Slide Deck for Talent Viewing

As a producer(admin) and client,

I want a cover slide to appear before viewing talents in the client decks tab

So that I can provide an introduction or branding before the talent details are displayed, offering a more polished and professional presentation

Points: 2

Assigned To: Jose Salazar, Fatemah Zare

Status: Complete

11. Make Spreadsheets Consistent Across All Tabs

As a producer(admin) and client,

I want all spreadsheets across different tabs to have the same functionality and appearance.

So that I can have a consistent and seamless experience when working with data, ensuring uniformity and reducing confusion

Points: 2

Assigned To: Alea Nablan

Status: Complete

Actual Screenshot:

The screenshot shows the 'Event 1' page with the following details:
event 1 description
Location : Ames, Iowa
Date : 2024-10-31T05:00:00.000Z
Category : Performing Arts

Navigation tabs: HOME, SUBMITTED DOCS, SELECTED DOCS, CLIENT DECKS (highlighted), FINALIZED DECKS, SUMMARY, EDIT FORM.

Select a client below to view their slide deck
client1
EXPAND DECK

Talents
Add Row | Save Data

Preference	Name	Gender	Height	Birth Date	Email
1	user123	Female	5' 2"	2001-10-26	user123@email.com
2	user234	Male	5' 10"	2000-05-03	user234@email.com
3	user345	Female	6' 0"	2005-05-03	user345@email.com

1-3 of 3

12. Test Suite Coverage and Passing Test Cases

As a software developer,

I want to ensure that all test suites and individual tests pass successfully, improving code reliability and coverage.

So that the application's functionality is verified, and any errors are caught early before deployment, ensuring quality control.

Points: 2

Assigned To: Shweta Kumaran, Alea Nablan

Status: Complete

13. Admin Event Creation

As a producer(admin)

I want to manually enter the start and end dates for an event in the text field without using the calendar picker tool.

So that I can have flexibility in date input and the page does not crash when manually typing the dates.

Points: 2

Assigned To: Rahul Baid

Status: Incomplete

14. CSS Files Need to Be Cleaned

As a software developer

I want to clean up redundant or empty CSS files and ensure the UI is consistent between deployments.

So that the application's UI matches the client's deployment or has consistent styling across environments

Points: 2

Assigned To: Rituparna Mandal

Status: Incomplete

15. Deploy the new application to Heroku

As a developer,

I want to deploy the application to Heroku,

so that it is accessible to users via the web and I can ensure the app works as expected in the production environment.

Points: 3

Assigned To: Rahul Baid

Status: Complete

Sprint 3:

16. Fix the Scrolling Issue on the Events Form Page

As a producer(admin),

I want to be able to scroll through the new events form page

So that I can complete and review all fields in the form without being restricted by the display area, improving form usability

Points: 1

Assigned To: Rahul Baid

Status: Complete

Actual Screenshot:

Event 1

event 1 description
Location : Ames, Iowa
Date : 2024-10-31T05:00:00.000Z
Category : Performing Arts

[BACK TO FORMS](#)

HOME SUBMITTED DOCS **SELECTED DOCS** CLIENT DECKS FINALIZED DECKS SUMMARY EDIT FORM

Use this page to create client-specific slide decks

Selected Users					
Name	Gender	Height	Birth Date	Email	State
user123	Female	5' 2"	2001-10-26	user123@email.c...	Texas
user234	Male	5' 10"	2000-05-03	user234@email.c...	Kentucky
user345	Female	6' 0"	2005-05-03	user345@email.c...	Louisiana

1-3 of 3 < >

Name

Gender
 Male
 Female
 Other

Height*
Height in Feet ' and Inches '"

Birth Date

17. RSpec Coverage on Legacy Model and Controllers Codebase

As a developer

I want to have >90% RSPEC coverage on the legacy codebase

So that we can guarantee the legacy controller logic is working as expected and new changes do not break the existing features.

Points: 2

Assigned To: Rituparna Mandal

Status: Complete

18. JEST Coverage on Legacy UI Code

As a developer

I want to have >90% Jest coverage on the legacy UI code

So that we can guarantee the UI logic is working as expected and new changes do not break the existing features.

Points: 2

Assigned To: Shweta Kumaran, Alea Nablan

Status: Complete

19. Fix CI Pipeline

As a developer

I want to have a CI pipeline that correctly tests the code before merged

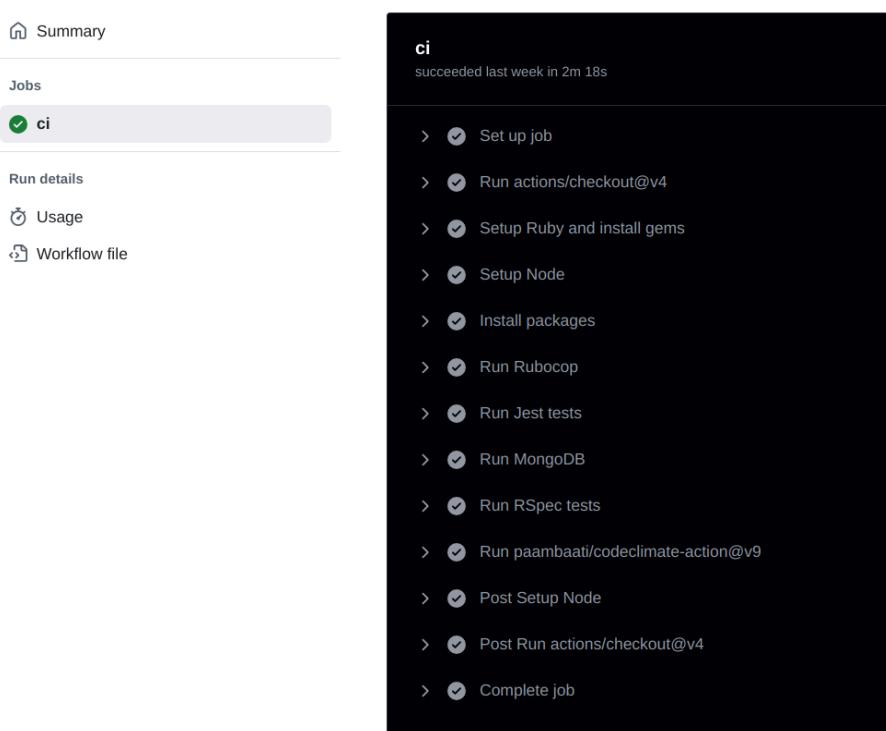
So that I do not have to spend resources on doing it manually and we can ensure new code does not break any existing cases

Points: 1

Assigned To: Rahul Baid

Status: Complete

Actual Screenshot:



20. Admin-Talent DM Functionality with selected multiple talents

As an event admin

I want to select multiple talents who have submitted forms for an event, open a chat window, and send a message to them.

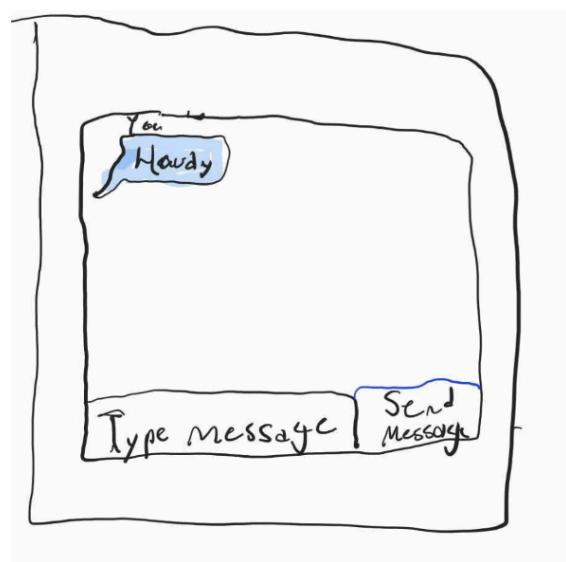
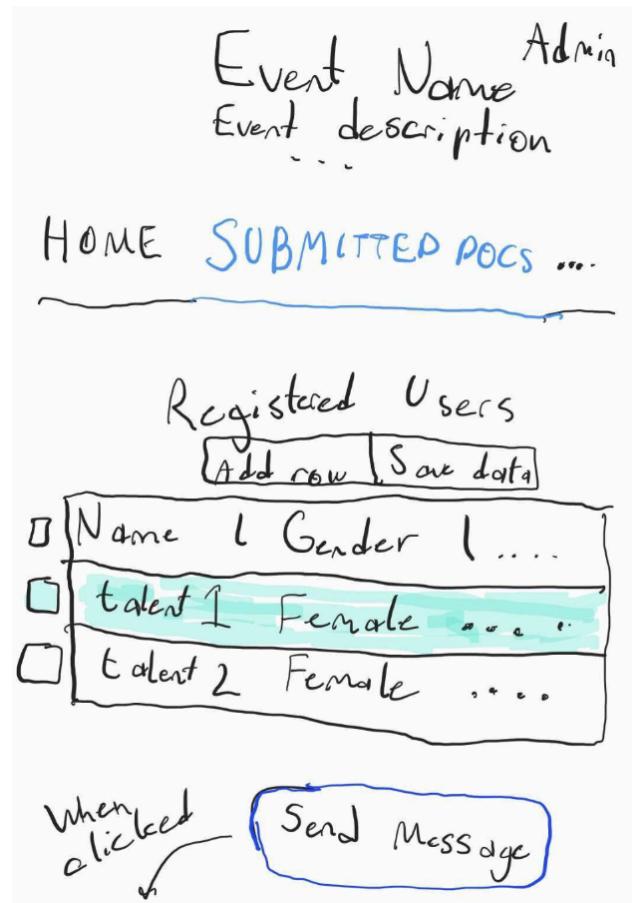
So that I can communicate event details and coordinate efficiently with the selected talents

Points: 2

Assigned To: Jose Salazar and Harsh Shah

Status: Complete

Lo-fi UI mockups/storyboards:



Actual Screenshot:

The screenshot shows a web application interface for managing user data. At the top, there is a header with the title "admin123" and some system information: "def", "Location : Hollywood, Florida", "Date : 2024-11-09T06:00:00.000Z", and "Category : Fashion". Below the header, there is a navigation bar with links: HOME, SUBMITTED DOCS (which is underlined), SELECTED DOCS, CLIENT DECKS, FINALIZED DECKS, SUMMARY, and EDIT FORM. A button labeled "BACK TO FORMS" is located in the top right corner. A message below the navigation bar says, "Use this page to create a master slide deck for this event." The main content area is titled "Registered Users" and contains a table with one row of data. The table has columns for Name, Gender, Birth Date, Email, State, and City. The first column has a checked checkbox. The data in the table is: Name: user123, Gender: (empty), Birth Date: 2024-11-03, Email: user123@gmail.c..., State: (empty), City: (empty). There are buttons for "CHAT WITH USER123", "Add Row", and "Save Data". At the bottom of the table, it says "1 row selected" and "1-1 of 1".

21. Triaging Slowness

As a developer

I want to see why the page renders slowly

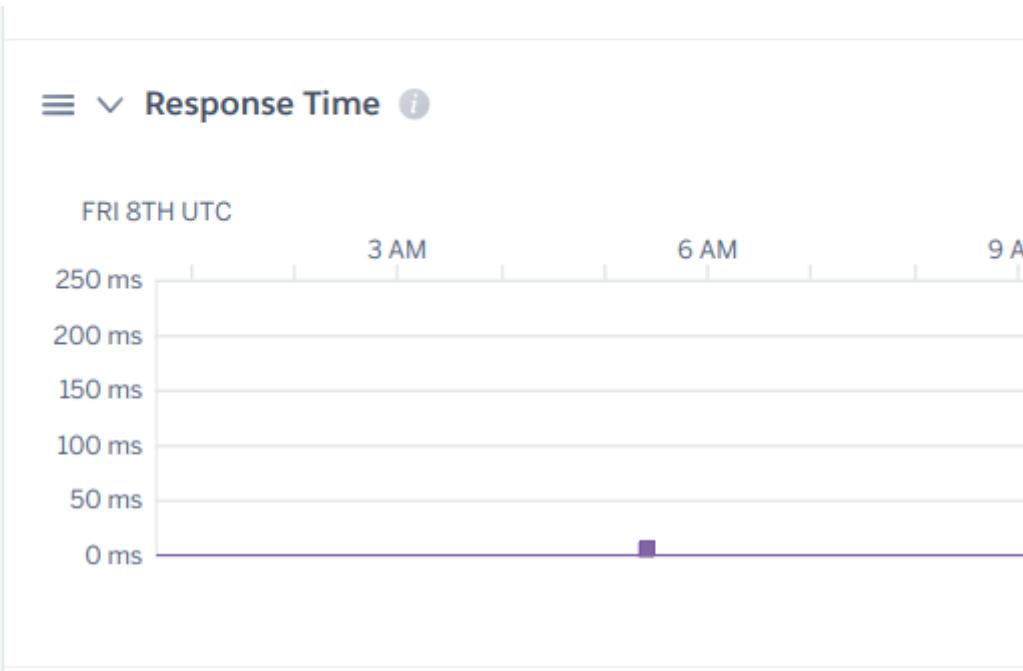
So that I can fix it for customers to not face the same issue

Points: 1

Assigned To: Rahul Baid

Status: Complete

Actual Screenshot:



22. Setup Cucumber

As a developer

I want to set up Cucumber for a react application

So that I can make and run tests to ensure the functionality of the application is present

Points: 2

Assigned To: Rahul Baid

Status: Complete

Sprint 4:

23. Admin-Talent Group DM with selected multiple talents

As an Event admin

I want to select multiple talents who have submitted forms for an event, open a chat window, and send a message to them.

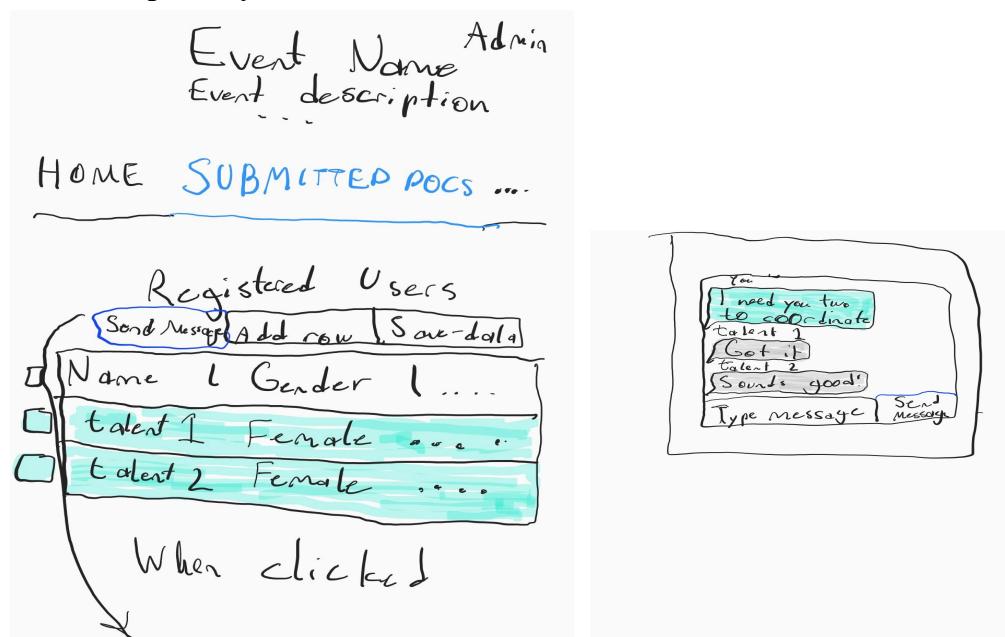
So that I can communicate and coordinate efficiently with the selected talents and receive their input.

Points: 2

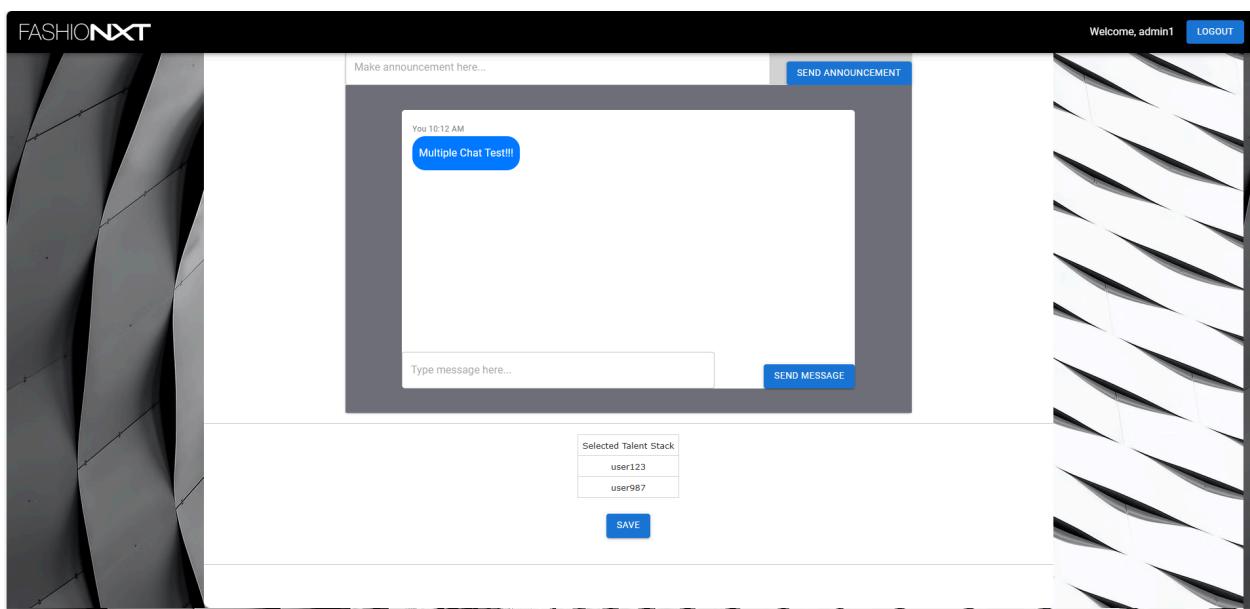
Assigned To: Jose Salazar

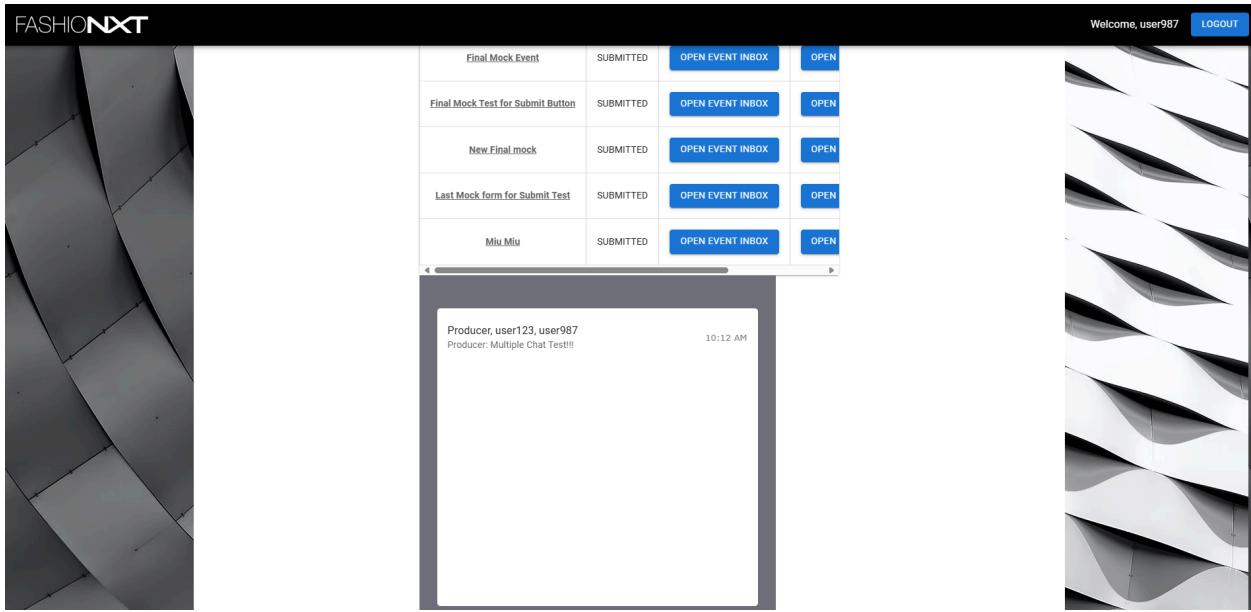
Status: Complete

Lo-fi UI mockups/storyboards:



Actual Screenshot:





24. Event 360 Login using OAuth

As a Events360 user,

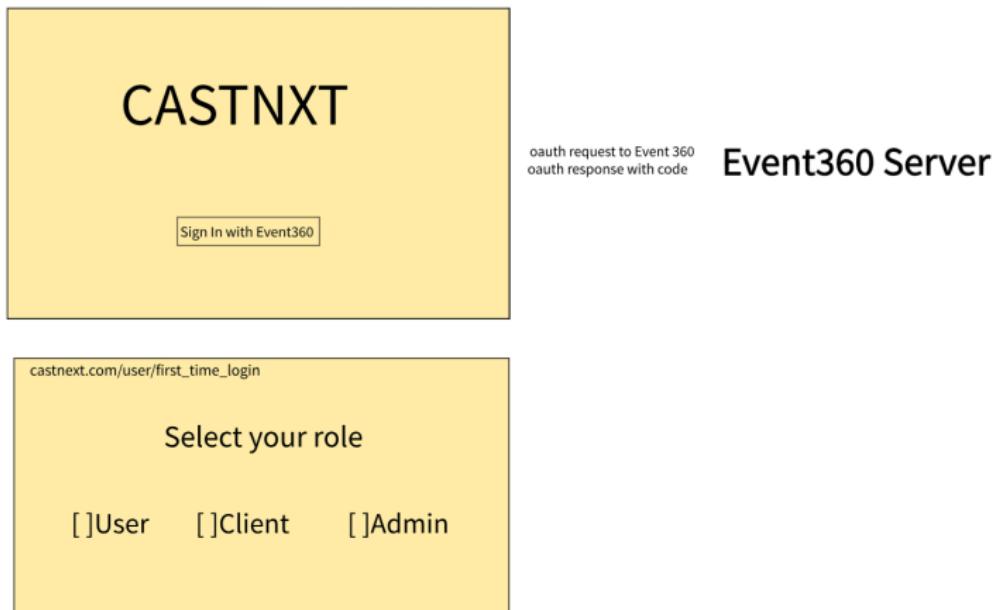
I should be able to use my Event360 account to access my CastNXT account
So that I do not have to maintain separate credentials

Points: 3

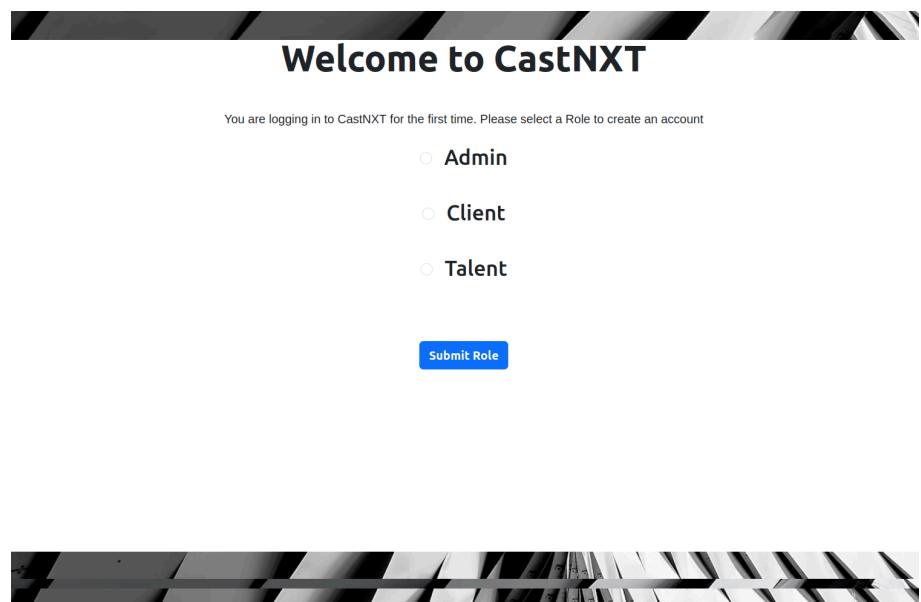
Assigned To: Rituparna Mandal

Status: Complete

Lo-fi UI mockups/storyboards:



Actual Screenshot:



25. Role Selection for New Users after OAuth Login

As a CastNXT User,
 I should be able to choose the role when I log in with Events360 for the first time.
 So that I can create an account in CastNXT for my role
 Points: 2

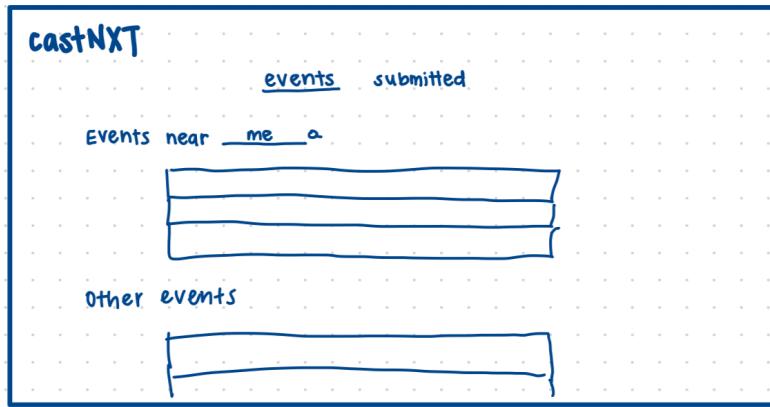
Assigned To: Rituparna Mandal
Status: Complete

26. Events Near Me

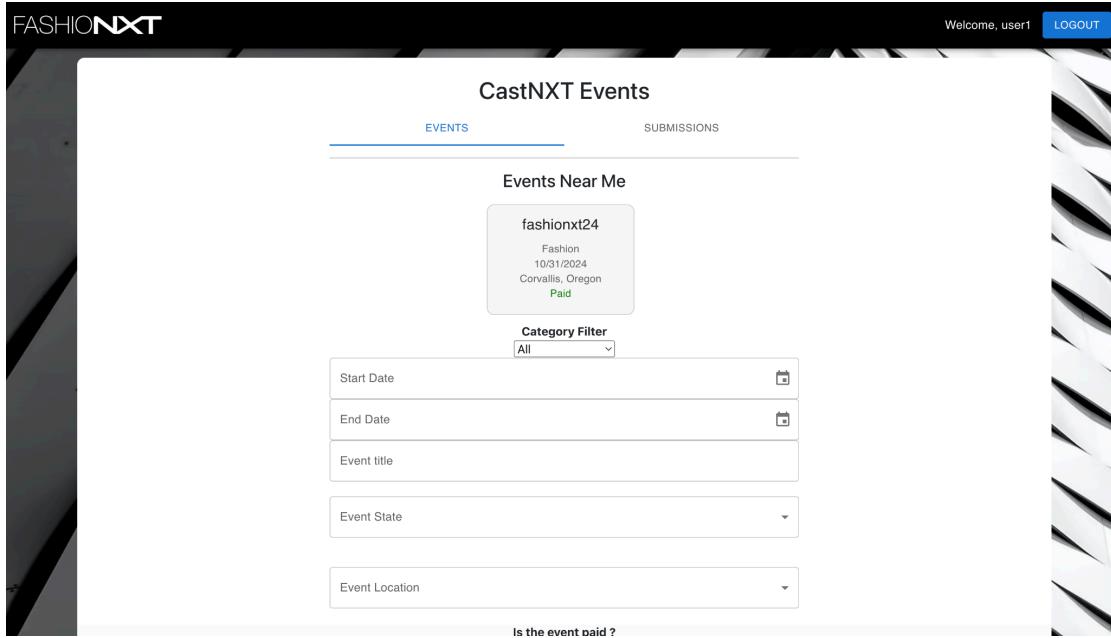
As an talent,
I want to see events near me.
So that I do not have to search for events in the city I am in
Points: 2
Assigned To: Shweta Kumaran
Status: Complete

Lo-fi UI mockups/storyboards:

talent has logged in



Actual Screenshot:



27. Spreadsheet with multiple filter capabilities

As an Event Admin

I want to filter through talents in the spreadsheets based on multiple features rather than using only one feature to filter.

So that I can view talents who have multiple specific features that I need.

Points: 2

Assigned To: Alea Nablan

Status: Complete

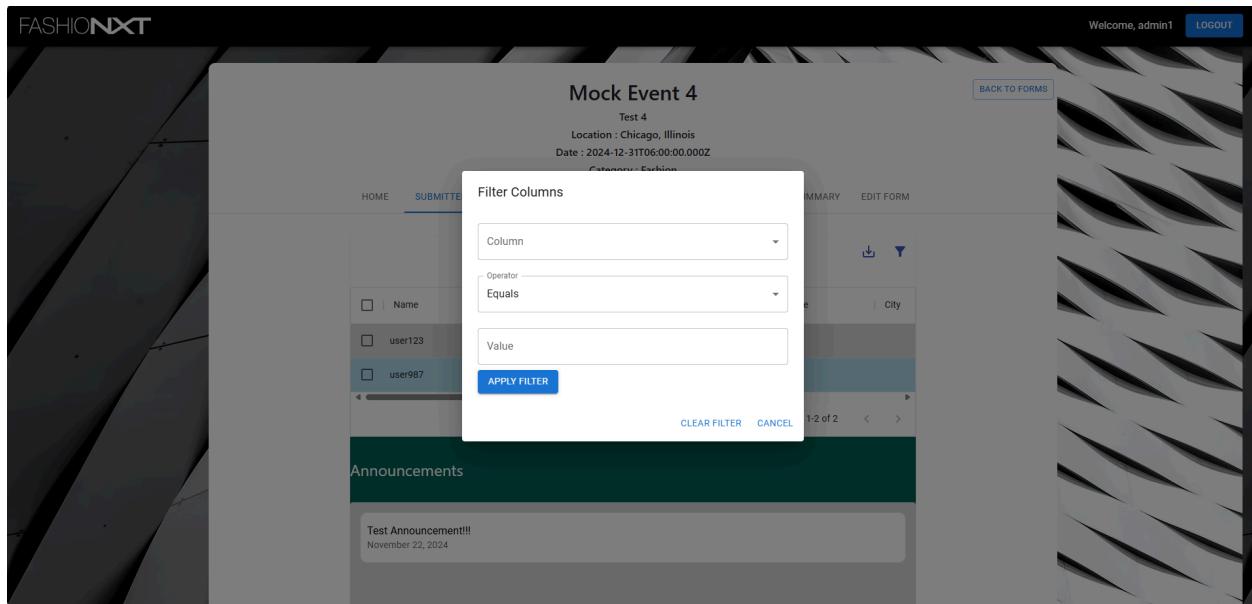
Lo-fi UI mockups/storyboards:

Name	Gender	Height	State
user 1	Female	5' 0"	Texas
user 2	male	6' 0"	Texas
user 3	female	5' 7"	Florida

Name	Gender	Height	State
user 1	female	5' 0"	Texas
user 2	male	6' 0"	Texas
user 3	female	5' 7"	Florida

Name	Gender	Height	State
user 1	female	5' 0"	Texas

Actual Screenshot:



28. Announcements

As an Event admin

I want to announce that all talents who submitted for an event see

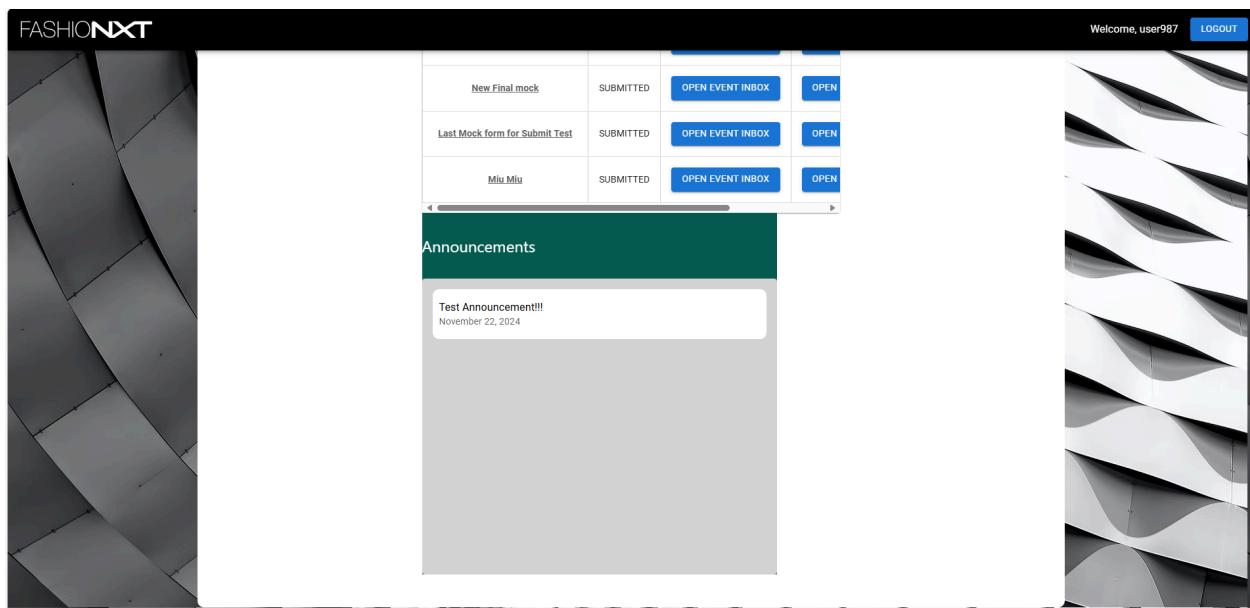
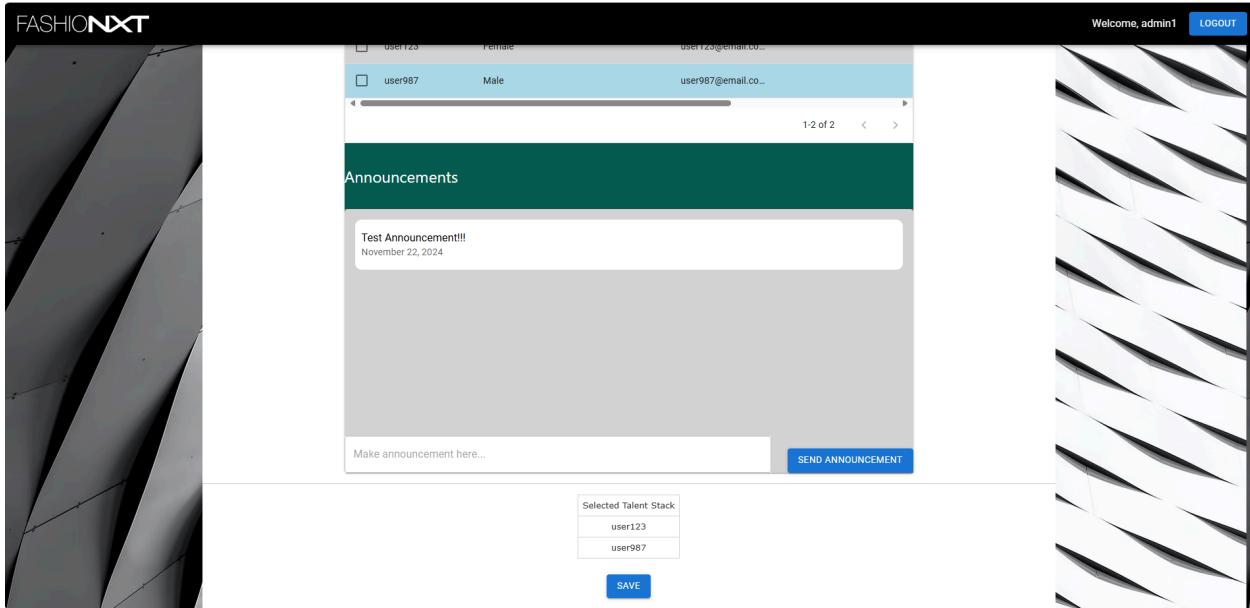
So that I can inform all talents of important event details/updates in a timely manner.

Points: 2

Assigned To: Jose Salazar

Status: Complete

Actual Screenshot:



29. Delete Slide

As an event admin
I want to delete the slides
So that I can remove the slides that I do not need anymore
Points: 2
Assigned To: Rahul Baid
Status: Complete

30. Image upload limit

As an event admin

I want to upload images up to 5MB in size.

So that I can avoid unintentionally uploading large images

Points: 1

Assigned To: Rahul Baid

Status: Incomplete

31. Submit Button Fix

As an admin user,

I want the form preview to indicate that it does not submit any data.

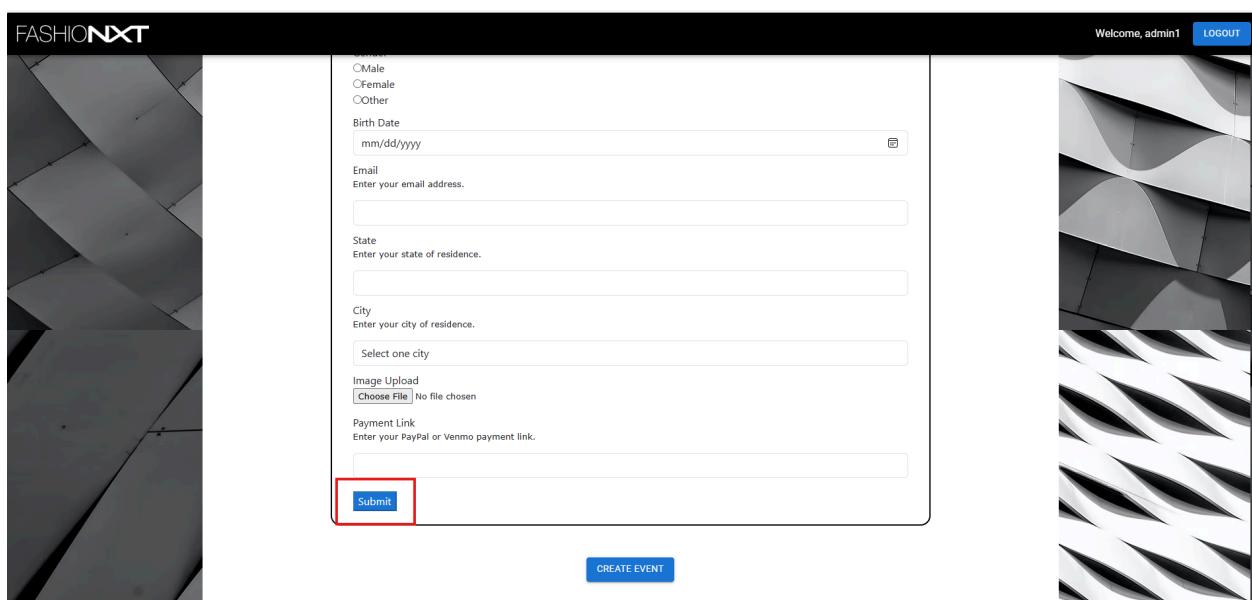
So that I understand it is only a preview and avoid confusion or mistaken submissions

Points: 2

Assigned To: Harsh Shah

Status: Completed

Actual Screenshot:



32. System Improvements

As a developer

I want to have faster builds and response times.

So that I can iterate quicker and improve user experience

Points: 2

Assigned To: Rahul Baid

Status: Complete

33. Cucumber for Admin

As a developer

I want the codebase to have comprehensive Cucumber scenarios.

So that modifying and adding new features does not unintentionally alter existing functionality

Points: 3

Assigned To: Rahul Baid, Alea Nablan

Status: Complete

34. Cucumber for User

As a developer

I want the codebase to have comprehensive Cucumber scenarios.

So that modifying and adding new features does not unintentionally alter existing functionality

Points: 2

Assigned To: Shweta Kumaran

Status: Complete

35. Cucumber for Client

As a developer

I want the codebase to have comprehensive Cucumber scenarios.

So that modifying and adding new features does not unintentionally alter existing functionality

Points: 2

Assigned To: Harsh Shah

Status: Complete

Understanding/Improving the Legacy Project

Learning the codebase:

Our process for understanding the legacy codebase was to look at the existing documentation to understand the intended flow of the application. When we managed to get the app set up locally, we used the app ourselves, pretending as though we were the users. This meant testing out and pressing every single button to see what the user sees.

Improving the codebase:

The main modifications made to the codebase were adding a CI pipeline and BDD set-up for Cucumber, and changing the deployment process. We changed the deployment process and automated it instead of requiring the deployment steps to be completed manually. We added a CI pipeline that runs linters and tests on branches which are a part of pull requests. This was to ensure that any new code that was pushed to the main did not fail any existing tests. As described later in this report, we set up Jest-Cucumber to be able to write Cucumber scenarios as part of the BDD process, which is something that did not originally exist in this repository.

Some of our stories were extensions of stories that were implemented by previous teams. We tried to make sure the code style and implementation approach were consistent with existing code (by either matching our style to theirs or vice versa) for the sake of code readability.

Team Roles:

For the most part, we have had a new person take the role of a Product Owner and Scrum Master each sprint. For Sprint 4, the same people served as Scrum Master and Product Owner again since the remaining two people in the team who had not served in a role were not actively working on the project. It can be assumed that all other members not listed in each row were developers during the Sprint.

Sprint #	Scrum Master	Product Owner
1	Jose Salazar	Shweta Kumaran
2	Harsh Shah	Alea Nablan
3	Rituparna Mandal	Rahul Baid
4	Rituparna Mandal	Rahul Baid

Iteration Accomplishments & Points:

Iteration	# of Stories Completed	# of Points Completed	Percentage Contribution

Iteration 1	12	39	35.13%
Iteration 2	13	27	24.32%
Iteration 3	10	15	13.51%
Iteration 4	15	30	27.02%
Total	50	111	100%

Team Member Contributions:

Sprint 1

Member	# of Stories Completed	# of Points Completed	Percentage Contribution
Rahul Baid	2	6	15.38%
Rituparna Mandal	1	4	10.25%
Harsh Shah	1	4	10.25%
Shweta Kumaran	2	4	10.25%
Alea Nablan	1	4	10.25%
Jose Salazar	3	9	23.07%
Milad Saadat	1	4	10.25%
Fatemah Zare	1	4	10.25%
Total	12	39	100%

Sprint 2

Member	# of Stories Completed	# of Points Completed	Percentage Contribution
Rahul Baid	1	3	11.11%

Rituparna Mandal	1	3	11.11%
Harsh Shah	2	3	11.11%
Shweta Kumaran	2	4	14.81%
Alea Nablan	2	4	14.81%
Jose Salazar	2	5	18.51%
Milad Saadat	2	3	11.11%
Fatemah Zare	1	2	7.40%
Total	13	27	100%

Sprint 3

Member	# of Stories Completed	# of Points Completed	Percentage Contribution
Rahul Baid	5	5	33.33%
Rituparna Mandal	1	2	13.33%
Harsh Shah	1	2	13.33%
Shweta Kumaran	1	2	13.33%
Alea Nablan	1	2	13.33%
Jose Salazar	1	2	13.33%
Milad Saadat	0	0	0%
Fatemah Zare	0	0	0%
Total	10	15	100%

Sprint 4

Member	# of Stories Completed	# of Points Completed	Percentage Contribution
Rahul Baid	4	7	23.33%
Rituparna Mandal	2	5	16.67%
Harsh Shah	3	5	16.67%
Shweta Kumaran	2	5	16.67%
Alea Nablan	2	4	13.33%
Jose Salazar	2	4	13.33%
Milad Saadat	0	0	0%
Fatemah Zare	0	0	0%
Total	15	30	100%

Total

Member	# of Stories Completed	# of Points Completed	Percentage Contribution
Rahul Baid	12	21	18.92%
Jose Salazar	8	20	18.02%
Shweta Kumaran	7	15	13.51%
Harsh Shah	7	14	12.61%
Rituparna Mandal	5	14	12.61%
Alea Nablan	6	14	12.61%
Milad Saadat	3	7	6.3%
Fatemah Zare	2	6	5.4%
Total	50	111	100%

Meetings with Client:

1. September 24, 2024, at 7 PM CST.

We plan to meet weekly at this time, depending on the client's weekly availability.

2. October 10, 2024, at 7 PM CST

We discussed logging in through Events360:

- Users can log in via the Events360 portal, enabling access to all applications.
- Terminology will be updated to Producer, Client, and Talent for clarity and consistency.
- Payments between Producers and Talents will occur through a master spreadsheet, where Producers can link their bank accounts.
- Producers can request Clients to submit files like music/logos via a common page for back-and-forth data sharing.
- A chat feature will be added next to the Client's name.
- Expand Deck will now show:
 - A cover slide is the first slide.
 - Model-specific slides are accessible by clicking the model name or navigating via the left/right arrows.
- Producers can add general instructions on the cover slide and use DM for communication.
- DM will show a history of messages, functioning like a Zoom chat.
- A DM feature will be added between Producers and Talents, enabling interactive communication on the model-specific slides (e.g., sending contracts or instructions).
- An attachment feature for images/documents will also be implemented.
- Consistency across tables: Include Talent details (6 columns with location, etc.).
- Features to add:
 - Sort (high/low), filter, and highlight rows/columns (10 colors max).
 - Download as CSV, including images.
- Use Google Sheets for inspiration for features and simplicity.
- AI will draft form ideas for casting events, offering Producers options for customization.
- AI-generated drafts will guide the creation of form elements (e.g., text boxes).
- Working on the AI implementation.
- Slides in the client deck should include a download button for CSV and slides.
- All slide-related views across tabs should follow a uniform design for usability and coding consistency.

- Add features like expanding the deck and CSV downloads to the finalized deck and event summary views.
- Implement a group DM feature for Producers to message selected Talents simultaneously (e.g., event instructions).
- Consider announcement capabilities for group communication without disrupting the app's functionality.

3. October 15, 2024, at 7 PM CST

- The website link will be simplified to castnxt.herokuapp.com.
- Single login will be implemented via Events360, ensuring all users access this app exclusively through Events360.
- Payment flow will remain simple:
 - Clicking the payment button redirects users to Venmo/PayPal.
 - Manager/Producer accounts can be connected to the app for seamless transactions.
 - Payment details will appear in the Summary tab, with a status update for completion of payment.
- The cover slide will serve as the main hub for general communication:
 - Top Half: Displays General Announcements (non-interactive).
 - Bottom Half: Hosts an interactive Zoom-like chat for exchanging general information (e.g., logos, photoshoot inspirations).
- Talent Slides will remain focused on individual talent, showing text and pictures without interactive elements.
- Fashionnxt LLC created an OpenAI API account.
- Multiple API keys will be generated to handle separate features.
- AI Features:
 - Form Creation: Use AI to suggest and draft forms for specific purposes (e.g., casting shows).
 - Talent Matching: Implement AI-driven matching based on criteria like body stats, looks, and qualifications.
- Investigate the CRM backend and its integration with the app.
- New Feature: Suggest matches based on criteria entered into the database.
 - Numeric fields: Allow tolerance filters (e.g., 10-20% variance for height, dress size).
 - Non-numeric fields: Allow filtering based on categorical fields (e.g., gender).
 - Users can choose specific columns to match in the pop-up table.
 - AI will assist by matching individuals based on inputs like appearance and qualifications.

- Download Feature: Ensure all spreadsheets are downloadable as CSV, including images.
- Multi-point filters will use equations/formulas to find matches based on producer queries.

4. October 22, 2024, at 7 PM CST

We demo the latest release to the clients. Major discussion points were:

- Increased timelines for AI form integration. We discussed some implementation details based on the research done in sprint 2.
- We also mentioned our priority of integration and unit test coverage of the legacy code this sprint. Hence, the client agreed to pause working on the AI feature mentioned in 1 and other features in the pipeline would be picked after testing is completed.
- Fine-grained the DM feature requirement with the client. We are working on a DM feature (phase 2) that involves the capability to message multiple talents at an event. The feature is discussed in user story 7.
- The client talked about future features like sign-sign-on and showed us a competitor platform for reference.

5. October 29, 2024, at 7 PM CST

We updated our progress for the sprint to the client. Major discussion points were :

- The client wanted some triaging to be done on the slowness of the app.
- The client had shared some bugs that he noticed earlier in the week. We went through those issues and prioritized them.
- We demoed some of the bug fixes for the bug reports made by the client.
- We discussed the UI button placements for the DM feature user story.

6. November 5, 2024, at 7 PM CST

We demoed the latest release for sprint 3 to the clients. Major discussion points were :

- We explained the slowness issue and upgraded the set-up to Heroku Dynos.
- Demoed the DM feature between Admin and Talents.
- We discussed features and possible UI changes for Sprint 4.

These included Event 360 login, Notifications, and Group Dm between Admin and Talent.

7. November 12, 2024, at 7 PM CST

We discussed the following points during the meeting:

- For spreadsheet multi-filtering, we discussed moving to Material UI Pro, which has built-in multi-filtering functionality. The link to pricing will be emailed to the client.
- Disabled the ability to repeatedly click the submit button to prevent duplicate submissions.
- Discussed adding a page for first-time users to select their role and be directed to the appropriate login window.
- Worked on restructuring the group DM and announcement features to support general group communication while retaining prior functionality. The target completion for both features is by Thursday or Friday.
- Confirmed that the group chat feature will support communication with multiple talents and clients, with a deadline aligned for completion.
- Working on the "Events Near Me" feature for talents, a test suite, and a new landing page with a default location set to Portland if the location is unspecified.
- Addressing bugs, such as the deletion of slides, enhancing the admin profile page UI, and refining functionalities like user deletion in submitted documents.
- Focused on merging changes and ensuring all bugs are resolved, with the latest link to be shared by Friday morning.
- Planned thorough testing of the entire application to identify and fix any remaining issues before Friday morning.

8. November 19, 2024, at 7 PM CST

We updated our progress for the sprint to the client. Major discussion points were :

- We demoed the MVP for sprint 4 and took the client's feedback.
- We discussed the issue of the App crashing in Heroku due to memory limits.
- We discussed the items to be added to the documentation for handing over the project to the next team.

9. November 26, 2024, at 7 PM CST

We discussed Heroku costs and memory usage:

- Reviewed the cost of Option 2 with the Heroku upgrade, which charges \$0.39/hr based on app runtime.
- Explained how costs are calculated and how scaling down dynos during inactivity can reduce costs.
- Highlighted that memory usage fluctuates due to dynamic libraries and app workload, with a maximum of 1 GB observed.
- Transitioning to JavaScript is possible but would require reimplementing many libraries.
- Suggested deferring this task to the next semester's team.
- Shared roadblocks encountered while working on the AI feature and suggested documenting these challenges for the next team.
- Discussed restructuring the code to improve efficiency and prepare for the next team.
- Proposed adding functionality to upload photos directly from the phone's camera.
- Encouraged leveraging current knowledge to resolve issues and leave thorough documentation for the next group.

BDD/TDD Details:

For the BDD process involving Jest, the team implemented user stories in the planning meeting to establish the behavior of any feature. These user stories were structured in a Cucumber structure. From here, the team implemented lo-fi prototypes. To implement the BDD process, some Jest test cases were implemented to ensure the behavior of the program was as expected. Through periodically building the test cases and the code, BDD was able to be implemented. This was beneficial for the team to identify any bugs with any features immediately to ensure proper behavior.

In terms of Jest-Cucumber, we faced numerous problems at the beginning. The previous team never had implemented Cucumber scenarios, which made it difficult to implement new scenarios. We tried to create a few scenarios initially for the existing behavior, however it did not work. Through research and collaboration with the TA, the team was able to identify the issues. Since this is a React project, it implements Material UI. The way Material UI implements HTML elements does not allow Cucumber to automatically detect it, which is why it failed. We implemented Jest-Cucumber with the use of a Selenium web driver for BDD and UI tests. This problem made it difficult to implement the BDD process with Cucumber.

We employed RSpec for TDD, focusing on testing individual models and controllers. The legacy code initially had less than 80% RSpec coverage. One of our sprint goals was to raise coverage to 90% by adding RSpec tests for all files. Additionally, new features that were developed were pushed with the corresponding RSpec scenarios added during each sprint. This was useful to

identify any controller and library level changes that were made during the sprints and writing the tests ensured that we covered all edge cases for the methods that we added.

shweta-kumaran/CastNXTfall24

Breakdown
270 FILES

Maintainability:

Test Coverage:

Codebase summary

Maintainability	Test Coverage
D 3 mos	A 90%

Repository stats

Code Smells	Duplications	Other Issues
74	184	0

```

config.eager_load is set to nil. Please update your config/environments/*.rb files accordingly:

* development - set it to false
* test - set it to false (unless you use a tool that preloads your test environment)
Open file in editor (cmd + click) `ue
/Users/rahulbaid/.rbenv/versions/2.6.6/lib/ruby/gems/2.6.0/gems/net-protocol-0.2.2/lib/net/protocol.rb:68: warning: already initialized constant Net::ProtocRetryError
/Users/rahulbaid/.rbenv/versions/2.6.6/lib/ruby/gems/2.6.0/gems/net-protocol-0.2.2/lib/net/protocol.rb:66: warning: previous definition of ProtocRetryError was here
/Users/rahulbaid/.rbenv/versions/2.6.6/lib/ruby/gems/2.6.0/gems/net-protocol-0.2.2/lib/net/protocol.rb:214: warning: already initialized constant Net::BufferedIO::BUFSIZE
/Users/rahulbaid/.rbenv/versions/2.6.6/lib/ruby/gems/2.6.0/gems/net-protocol-0.2.2/lib/net/protocol.rb:206: warning: previous definition of BUFSIZE was here
/Users/rahulbaid/.rbenv/versions/2.6.6/lib/ruby/gems/2.6.0/gems/net-protocol-0.2.2/lib/net/protocol.rb:541: warning: already initialized constant Net::NetPrivate::Socket
/Users/rahulbaid/.rbenv/versions/2.6.6/lib/ruby/2.6.0/net/protocol.rb:503: warning: previous definition of Socket was here
.6740d66978e699121cc64a62
.....step 1
step 2
step 3
eventtest_user@gmail.com
test_events
step 4
.eventtest_user@gmail.com
test_events
step 1
.....tests
{"email":>"hometest_admin@gmail.com", "name":>"hometest_admin", "password":>"12345678", "type":>"ADMIN", "controller":>"home", "action":>"signup"}
Unexpected error
{"email":>"hometest_admin@gmail.com", "name":>"hometest_admin", "password":>"12345678", "type":>"ADMIN", "controller":>"home", "action":>"signup"}
{"email":>"hometest_client@gmail.com", "name":>"hometest_client", "password":>"12345678", "type":>"CLIENT", "controller":>"home", "action":>"signup"}
{"email":>"hometest_talent@gmail.com", "name":>"hometest_talent", "password":>"12345678", "type":>"USER", "controller":>"home", "action":>"signup"}
{"email":>"hometest_admin@gmail.com", "name":>"hometest_admin", "password":>"12345678", "type":>"ADMIN", "controller":>"home", "action":>"signup"}
{"email":>"hometest_admin@gmail.com", "name":>"hometest_admin", "password":>"12345678", "type":>"ADMIN", "controller":>"home", "action":>"signup"}
.....Auth data: test@example.com
User successfully created: #<User:0bjectId('6740d66978e699121cc64a62'), created_at: Fri, 22 Nov 2024 19:07:24 UTC +00:00, email: "test@example.com", name: "Test User", dated_at: Fri, 22 Nov 2024 19:07:24 UTC +00:00, user_type: "new_user"
User successfully retrieved from DB: #<User:0bjectId('6740d66978e699121cc64a62'), created_at: Fri, 22 Nov 2024 19:07:24 UTC +00:00, email: "test@example.com", name: "Test User", updated_at: Fri, 22 Nov 2024 19:07:24 UTC +00:00, user_type: "new_user">
Auth data: test@example.com
.....6740d66978e699121cc64a62
{"curated":>"false", "formData":>"{"name":>"aaaa", "email":>"aaaa@gmail.com", "talentName":>"aaaa", "state":>"Kentucky", "city":>"Ames", "paymentLink":>"paypal.me/randomlink"}, "slide_updated":>"true", "slide_id":>"6740d66978e699121cc64a62", "slide_order":>1}
.....Auth data: #<OmniAuth::AuthHash info=>#<OmniAuth::AuthHash::InfoHash email="testuser@example.com"> provider="events360" uid="123456">
Auth data: #<OmniAuth::AuthHash info=>#<OmniAuth::AuthHash::InfoHash email="testuser@example.com"> provider="events360" uid="123456">
Auth data: #<OmniAuth::AuthHash info=>#<OmniAuth::AuthHash::InfoHash email="testuser@example.com"> provider="other_provider" uid="123456">
Auth data: testuser@example.com
{"name":>"hometest_admin", "email":>"hometest_admin@gmail.com", "password":>"12345678", "type":>"ADMIN", "controller":>"home", "action":>"signup"}}
...
Finished in 4.36 seconds (files took 2.57 seconds to load)
85 examples, 0 failures
JSON Coverage report generated for RSpec to /Users/rahulbaid/Desktop/CastNXTfall24/coverage. 839 / 918 LOC (91.39%) covered.

```

```

yarn run v1.22.22
$ jest --updateSnapshot --forceExit --silent --coverage
PASS spec/javascript/Components/Admin/AdminUserTable.spec.js
PASS spec/javascript/Components/Client/ClientEventSummary.spec.js
PASS spec/javascript/Components/Admin/AdminEventSummary.spec.js
PASS spec/javascript/Components/Client/ClientEventFeedback.spec.js
PASS spec/javascript/Components/Client/ClientHomepage.spec.js
PASS spec/javascript/Components/Admin/AdminEventHome.spec.js
PASS spec/javascript/Components/Shared/DatePicker.spec.js
PASS spec/javascript/Components/Admin/AdminHomepage.spec.js
PASS spec/javascript/Components/Navbar/Header.spec.js
PASS spec/javascript/Components/Client/ClientEventPage.spec.js
PASS spec/javascript/Components/Forms/FormBuilder.spec.js
PASS spec/javascript/Components/Filter/LocationFilter.spec.js
PASS spec/javascript/Components/Admin/AdminSubmittedDocs.spec.js
PASS spec/javascript/Components/Filter/ColumnFilter.spec.js
PASS spec/javascript/Components/Filter/CategoryFilter.spec.js
PASS spec/javascript/Components/Filter/IsPaidFilter.spec.js
PASS spec/features/step_definitions/client_homepage.steps.js
PASS spec/features/step_definitions/user_homepage.steps.js
PASS spec/features/step_definitions/admin_homepage.steps.js
PASS spec/javascript/Components/User/UserHomepage.spec.js (5.298 s)
PASS spec/features/step_definitions/admin_events.steps.js (5.388 s)

File          %Stmts %Branch %Funcs %Lines Uncovered Line #
ALL files    89.37 75.59 75.7 90.74
components/Admin
  AdminEventHome.js      84.09 69.87 67.56 86.2
  AdminEventSummary.js   89.52 72.72 78.57 89.89 114,181-183,194,229-264
  AdminHomepage.js       93.75 83.33 83.33 93.75 41
  AdminSubmittedDocs.js  95.23 80     100    95    31
  AdminUserTable.js      78.5   65.11 56.71 81.81 ...,185-193,212,265,271,285-287,305-310,337-342,379,396,452-457,474-706
components/Client
  ClientEventFeedback.js 89.39 60     74.07 89.23 128,165,254-416,470-474
  ClientEventPage.js     100    100    100    100
  ClientEventSummary.js  92.5   83.33 81.81 92    105,132,199-204
  ClientHomepage.js      90.9   75     80     90.9  37
components/Filter
  CategoryFilter.js      100    100    100    100
  ColumnFilter.js        100    75     100    100  47
  IsPaidFilter.js        100    100    100    100
  LocationFilter.js      100    100    100    100
components/Forms
  FormBuilder.js         96.87 100    87.5   96.66 83
  FormBuilder.js         96.87 100    87.5   96.66
components/Navbar
  Header.js              100    100    100    100
  Header.js              100    100    100    100
components/Shared
  DatePicker.js          92.72 100    77.77 92.72
  DatePickerEnd.js        88.88 100    66.66 88.88 60-61
  DatePickerStart.js     94.44 100    83.33 94.44 58
  DatePickerStart.js     94.73 100    83.33 94.73 60
components/User
  UserHomepage.js         90.58 81.63 73.68 92.68
  UserHomepage.js         90.58 81.63 73.68 92.68 227,275,339,379-389,403-406,452,485,602,746
utils
  AdminDataUtils.js       97.91 85.71 90     100
  DataParser.js           100    80     100    100  3-12
  FormsUtils.js           100    100    100    100
  RangeFilter.js          87.5   100    50     100

Test Suites: 21 passed, 21 total
Tests:       147 passed, 147 total
Snapshots:   30 passed, 30 total
Time:        5.836 s, estimated 8 s
Force exiting Jest: Have you considered using `--detectOpenHandles` to detect async operations that kept running after all tests finished?
↑ Done in 6.20s.

```

Configuration Management:

Our configuration management approach is as follows:

1. We created a new branch for every feature, spike, patch, and triage, and would make the appropriate changes or enhancements there.
2. Once a feature/enhancement has been completed on the branch, to merge the changes to the main branch, a pull request needs to be made. At least 1 other person must review this request and approve it before changes are made.

We had 4 releases in total and shipped a new version (release) of the application at the end of each sprint.

We had 3 spikes throughout this project.

1. AI Form Generator: The client wanted to add an AI component to the project, where an admin could ask AI (imagine ChatGPT) in the event creation page to generate a form for an event as specified by the admin. A spike was done to determine how the API of an existing LLM can be integrated into our application with code, researching different LLMs that can be used, and how to make sure each user has a different session. We completed the spike, but due to other higher-priority stories, did not get to implement this.
2. Filtering by Multiple Features: One of our user stories in Sprint 4 was to extend the filtering capabilities to allow filtering by multiple features/categories. However, figuring out how to do this wasn't immediately clear. We used Material UI to filter by one category, but using it to filter by multiple categories would require a Pro subscription. Alternative, free approaches to multi-point filtering were investigated to see if Material UI Pro was necessary or not. We ended up not needing to use Material UI Pro.
3. Triaging Slowness: An issue that was raised by our client in one of our meetings is that when navigating to the application, the page would buffer for a long time before loading. This makes the app "unusable" since people will not want to wait that long every time they visit the app to use it. We were tasked with finding the cause of this issue and coming up with solutions. The cause of the issue is related to our Heroku dynos. Since our application was previously using Eco Dynos on Heroku, after ~30 minutes, the application would become idle, and when used again afterward, would take a while for the application server to start up. The solution we proposed was to upgrade the application to Basic Dynos to minimize the slowness.

Heroku Production Release Issues:

2 main issues occurred when our application was deployed to Heroku.

1. As mentioned earlier in this report, the Heroku application would take a while to load before it could be used. The issue turned out to be because the application server would become idle after a certain amount of time. This idle timeout is a result of having Eco Dynos for our application. Basic Dynos was purchased instead which resolved the issue.
2. Another issue we noticed is that the current image size of the app is larger than the limit of Eco/Basic Dynos.

During Sprint 4, we noticed that the Heroku deployment showed errors of "App Crashed" occasionally. On investigation, we found that the crash was due to a memory limit exceeded error (R15 crash). We have discussed this issue with Professor Ritchey and the Client.

One possible solution is to upgrade the Heroku deployment to one that has higher memory (the current quota is 500 MB). Another suggestion is to decouple the front-end

and back-end and deploy them separately. This will need a significant amount of code refactoring.

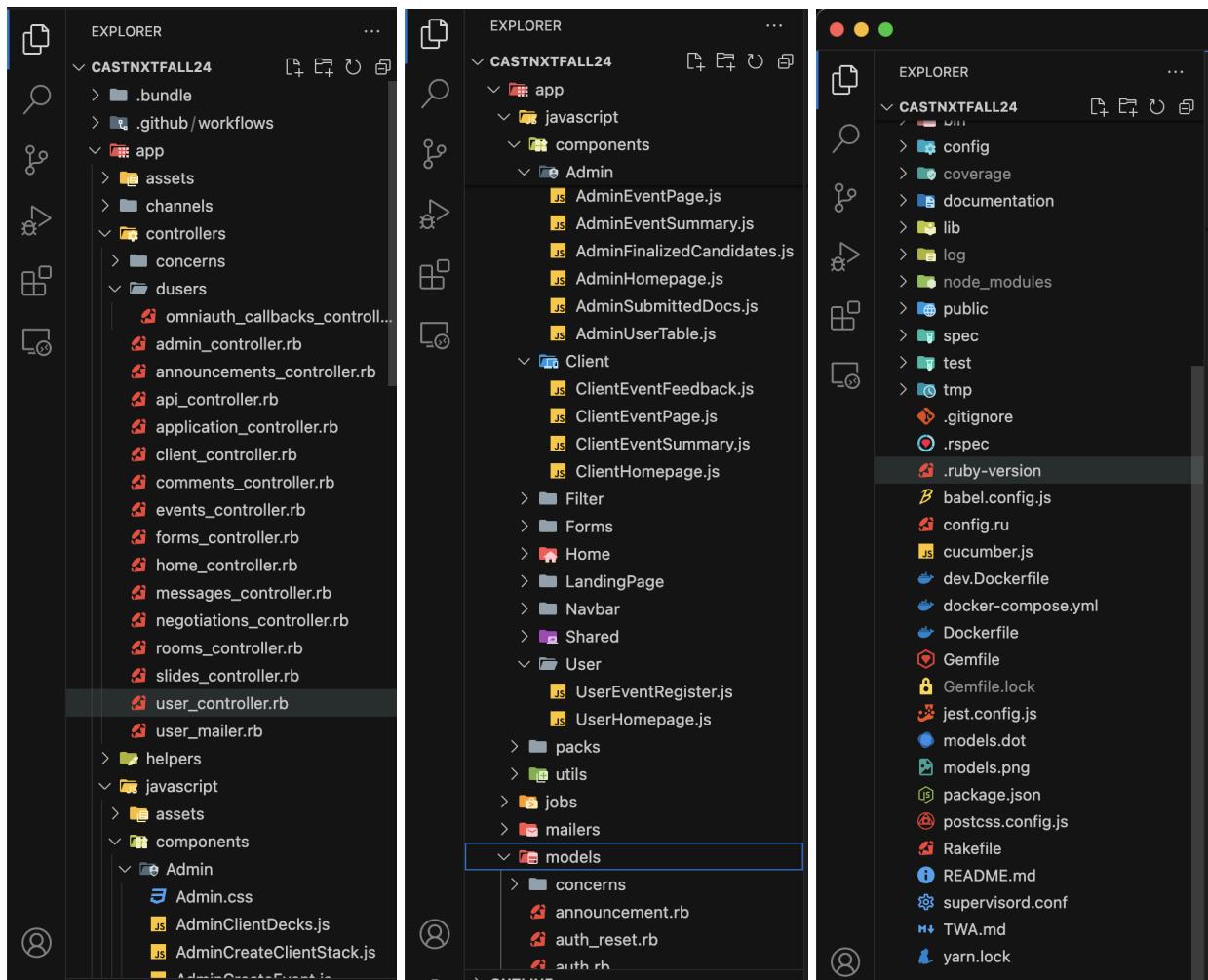
Gems/Tools used in application:

Tool/Gems	Description	Benefits or problems
GitHub	Version control software to share and collaborate on code	Easy to use, offers large set of features which are useful for configuration and project management (GitHub Actions, GitHub Projects)
Code Climate	Platform which analyzes the code quality of a repository, providing a simple score for it along with issues to improve on	Easy to understand results on webpage and provides actionable feedback on what can be improved in repository
mongoid (gem)	The official ODM (Object-Document Mapper) framework for MongoDB in Ruby	easy to use + good documentation
webpacker (gem)	Used to compile JS scripts and other assets	Very difficult to work with => poor error messages, it's already deprecated
omniauth (gem)	Ruby gem which handles Oauth	
Jest-cucumber (npm)	Cucumber.js alternative which allows writing Cucumber tests on top of Jest (with scenarios in the form of Given, When, Then)	Helped us to run cucumber/UI tests with the help of Selenium Webdriver + good documentation
Selenium Webdriver (npm)	Browser automation library, which was used to run Cucumber scenarios	
SimpleCov	Code coverage reporting tool for Ruby.	Provides a clean API for displaying the results of Ruby

		test results and coverage
React (npm)	A JavaScript framework we used to design the front-end of the application. Used the Material UI Library for the JavaScript components it provides	Consumes a large amount of memory, plus React was difficult to upgrade in a way that kept our application functional

Repo Contents & Deployment Process:

GitHub Repo:



Deployment Process:

We used GitHub Actions to deploy our application. The file that is being used for deployment is the `deploy.yaml` file by GitHub Actions (Located in `.github` folder/workflows). To deploy, you will first want to create a Heroku application for the CastNXT application and make sure you

have Heroku CLI set up. Modify the deploy.yaml file accordingly, and then push these changes to the main branch of the GitHub repository being used for CastNXT. If you want to change some of the technical steps being used for deployment, edit the Dockerfile. Other configuration settings can be found in the config folder.

Another necessary part of the application will be to connect it to a MongoDB cluster. If you need to do this yourself, then follow the following steps:

1. Create a Free MongoDB Atlas Cluster
2. Get MongoDB Atlas Connection URL
3. Connect Heroku Deployment to MongoDB cluster created, by editing the config variables in the Heroku app created. It should be in the form of a key-value pair, where the key is “MONGODB_URI”, and the value is the connection URL string obtained from MongoDB. Make sure to add the database name in the URL before you connect.

Important Links:

1. GitHub repository: <https://github.com/shweta-kumaran/CastNXTfall24>
2. GitHub Projects: <https://github.com/users/shweta-kumaran/projects/2>
3. Heroku Deployment: <https://castnxtfall2024-a6422e600193.herokuapp.com/>
4. Presentation and Demo Video Merged:
<https://drive.google.com/file/d/1U3UYcFPb63iHnqjsuZD4pQ2aEq0gYNCf/view?usp=sharing>
5. Presentation Video:
<https://drive.google.com/file/d/1iJmyHzgQ0E1t8oS6MaDnyIIjwEb4usc/view?usp=sharing>
6. Demo Video:
<https://drive.google.com/file/d/1r54dxhGgj5BG2X4f0Cwe8Ugkj8-fUYj8/view?usp=sharing>