

# **CSCE 606: SOFTWARE ENGINEERING**

## **CAST NXT Final Report**

### **Summary:**

CastNXT is a web-based platform designed to automate the process of talent casting for various events such as fashion shows and plays. The system involves three key stakeholders: Talent, Producer, and Client. As a user, the Talent persona can sign up for different events and track the progress of their application. The Producer, who is responsible for managing the event, can select talent, create lists of performers for clients, eliminate candidates from the selection pool, and negotiate talent decks. The Client can select their preferred performers and negotiate based on their choices.

The main contact person for CastNXT is Tito Chowdhary, the owner of FashioNXT, while the stakeholders include FashioNXT, event coordinators, models, and designers. The primary customer need is to streamline the talent casting and client negotiation process to automate event management. CastNXT addresses this need by providing a common platform for all stakeholders to automate the event management process, including creating lists of applicants for clients, shortlisting people for events, and facilitating back-and-forth negotiations with clients.

### **All user stories:**

#### **1) Deploying the application to Heroku - Anshul Mendiratta**

**Story:** The link for Heroku left by the last team wasn't operational. Further, the code in its current state was producing errors. We set up MongoDB and Heroku and fixed the code to get it up and running

**Point:** 3

**Iteration:** 0-1

**Status:** Done

#### **2) Developing a docker file for running the application locally - Manaswini Devabhaktuni**

Docker file is developed to easily run the application locally irrespective of the environment we have. We can run the application with a single command after cloning the file. This is the story which has taken in client knowledge to simplify the process of running the application locally.

**Point:** 5

**Iteration:** 0-1

**Status:** Done

### **3) Adding validations to the fields in UI at multiple locations - Manaswini Devabhakthuni**

- i) Admin is able to create events with past dates - Validation added to fix this
- ii) Admin is able to create events without providing any details - Validation added to fix this
- iii) User is able to register past events - Validation added to fix this
- iv) User is able to register events without giving any details - Validation added to fix this

**Point:** 3

**Iteration:** 2

**Status:** Done

### **4) Fixed issue with webpack hot reload in docker - Manaswini Devabhakthuni**

Docker file is developed to easily run the application locally irrespective of the environment we have. We can run the application with a single command after cloning the file. This is the story which has taken in client knowledge to simplify the process of running the application locally.

**Point:** 3

**Iteration:** 2

**Status:** Done

### **5) Added functionality for the admin to be able to edit events - Anshul Mendiratta**

Once the admin had created an event, there was no way to update the details about the event. We have added the functionality for the admin to be able to do that.

**Point:** 3

**Iteration:** 2

**Status:** Done

### **6) Added functionality to reset password - Kelvin Wang**

If the user forgot their password, there is no way for the user to log in aside from admin help. This will be replaced with a SAML system in the future as discussed in subsequent client meetings.

Database credentials for the email system were leaked and the API key for this was replaced and fixed.

**Point:** 3

**Iteration:** 2

**Status:** Done

## 7) Payment gateway integration - Amalesh Arivanan

The user should pay for certain tiers of services, Stripe integration is being added to facilitate this need.

**Point:** 3

**Iteration:** 2

**Status:** Done

## 8) Added Email Notifications - Anshul Mendiratta

Added functionality to be able to send emails to different users in different scenarios. These emails may be sent out either -

- a) During sign-up for email verification
- b) Send notifications to the user when an event (For example, someone registers for the event) happens

**Point:** 3

**Iteration:** 3

**Status:** Done

## 9) Fixed Date Picker Crushing Problem - Ming Li

When the admin users try to create an event, there are 2 problems that might occur:

- a) The date picker will cause the whole web page to crash by typing any number
- b) The user can not delete the input date time, “01/01/1971” will show up every time the users attempt to delete the input date time

These problems have been solved. Now the admin users can type the date time themselves or select the date time from the picker. And he can retype or re-select the date time by simple deletion of the input time.

**Point:** 3

**Iteration:** 3

**Status:** Done

## **10) Fixed User Event Register Problem - Ming Li**

Users can not register for existing events, even if they have input the correct name and email. There will always be an Error notice saying they did not input the name or email.

This problem has been solved. Now the users can successfully register for exciting events if they have input their name and email. On the contrary, they can not register if either of these two fields is blank.

**Point:** 1

**Iteration:** 3

**Status:** Done

## **11)Created New Landing Page for CASTNXT App - Manaswini Devabhakthuni**

Created a new landing page and integrated it into the existing cast nxt application.CastNXT wants to implement single sign-on and to land on the home page when clicking on the Cast nxt app it requires a landing page

i) As a user/admin/client

I want to see the Cast nxt landing page when I click on cast next after sign-on

So that I can see all the details of cast nxt events details and access the cast nxt app

**Point:** 3

**Iteration:** 3

**Status:** Done

## **12) Added/Fixed Ruby Credentials Manager for CastNXT - Kelvin Wang**

Fixing and migrating exposed secrets into secured storage so that a leak of credentials never occurs again, like the SendBlue hack which occurred with the project prior to our team's takeover.

**Point:** 3

**Iteration:** 3

**Status:** Done

## **13) Adding Site Admin Panel for Tito to Seamlessly Manage the App - Kelvin Wang**

Tito should not need to go into the code and make changes every time he wants something to change, for instance, the cut his company makes for each transaction between admins and talents. These figures will be able to be modified by Tito, himself without external intervention.

**Point:** 3

**Iteration:** 3

**Status:** Done

**14) Transfer of Heroku and Github accounts for all teams in Events360 to Client-  
Anshul Mendiratta**

All the projects under Events360 were associated with individual student Heroku and Github accounts. This caused issues once new teams came in since everything had to be set up again. To prevent this a Heroku and Github account for the client was created and all the projects were migrated to those accounts.

**Point:** 3

**Iteration:** 4

**Status:** Done

**15) Added user logins, Event details, Navbar, and Footer on the landing page- Manaswini  
Devabhakthuni**

As a user when I land on the landing page

I want to have options to log in as Admin/Client/User and have Navbar, Event details

So that I am directed to any one of Admin/User/Login and have a footer and Navbar

**Point:** 3

**Iteration:** 4

**Status:** Done

**16) Solved the Create Event crashing problem for admin users - Ming Li**

The Create New Event function for the admin users would be invalid when changing an event status, a click on that button will lead to the crash of the whole page, which can not be solved by refreshing the page or re-login.

This problem has been solved, any changing of event status now won't affect the Create New Event function or the whole system.

**Point:** 2

**Iteration:** 4

**Status:** Done

**17) Solved the event name disappearing - Ming Li**

The event name and category would directly disappear when changing that event status, which can not be solved by refreshing the page or re-login.

This problem has been solved, now changing of event status won't affect the display of the event name or category.

**Point:** 3

**Iteration:** 4

**Status:** Done

#### **18) Implemented chat functionality between the Producer and the Client - Anshul Mendiratta**

The client wanted the ability for the producer and client to be able to make suggestions/client for each talent slide. A chat functionality for every slide was added for this purpose.

**Point:** 3

**Iteration:** 5

**Status:** Done

#### **19) Implemented a new landing page similar to the EventNxt landing page - Manaswini Devabhakthuni**

As a user of the app,I want my landing page similar to all other nxt landing pages

So that the user feels he is in the same domain.

**Point:** 3

**Iteration:** 5

**Status:** Done

#### **20) Added All States All Cities to the Filter of User Page - Ming Li**

Keep the option visible for All States and All Cities. So the user can revert back to making no choices, hence the user can see all the forms of that category.

**Point:** 3

**Iteration:** 5

**Status:** Done

#### **21) Implemented Date and key-word Filter to User page -Ming Li**

Added filter options Event Date and Event name key-words. Now users can filter the events based on the potential start and end dates. Now users can filter the events based on the keywords or key characters in the event title, users don't have to provide the exact title name.

**Point:** 3

**Iteration:** 5

**Status:** Done

## **22) Made users visible to the submitted events - Ming Li**

Made users visible to the submitted events, and those events that have passed won't be continually stayed on their page. Moreover, the status of events is also visible to the users.

**Point:** 3

**Iteration:** 5

**Status:** Done

## **23) 3 User Type to UnifiedUser Database schema Merge Event - Kelvin Wang**

Merged all 3 types of users into a single type such that any user can be any of the types of users at the same time in the same account. Massive changes to all controllers and models were required, this change is a major version breaking-type change.

**Point:** 3

**Iteration:** 5

**Status:** Done

## **24) Frontend update - Kelvin Wang**

Updated the front end to reflect changes to the Talent, Admin, Client -> UnifiedUser schema.

**Point:** 3

**Iteration:** 5

**Status:** Done

## **25) Payment update - Amalesh Arivanan**

Worked on adding a 'payments' option to facilitate client-talent transactions

**Point:** 2

**Iteration:** 5

**Status:** Done

## **Lo-Fi UI Mockups :**

FashioNXT

User   Admin   Client

Username

Email address

Password

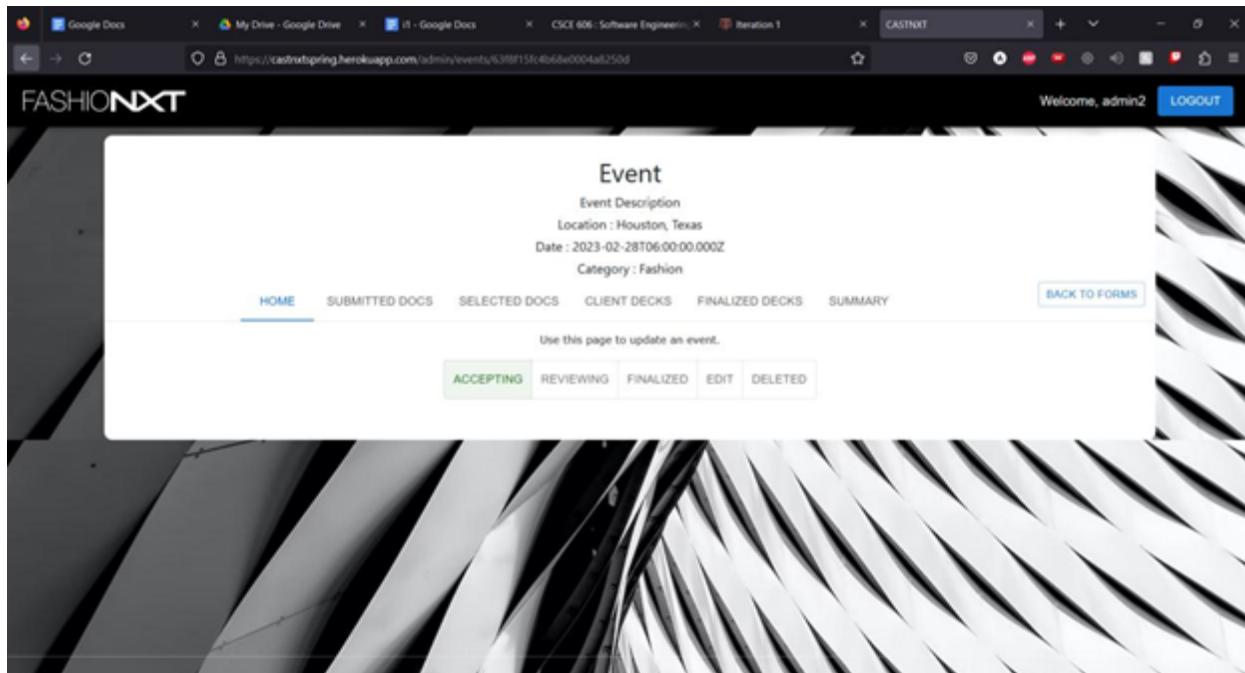
Forgot Password?

Event List

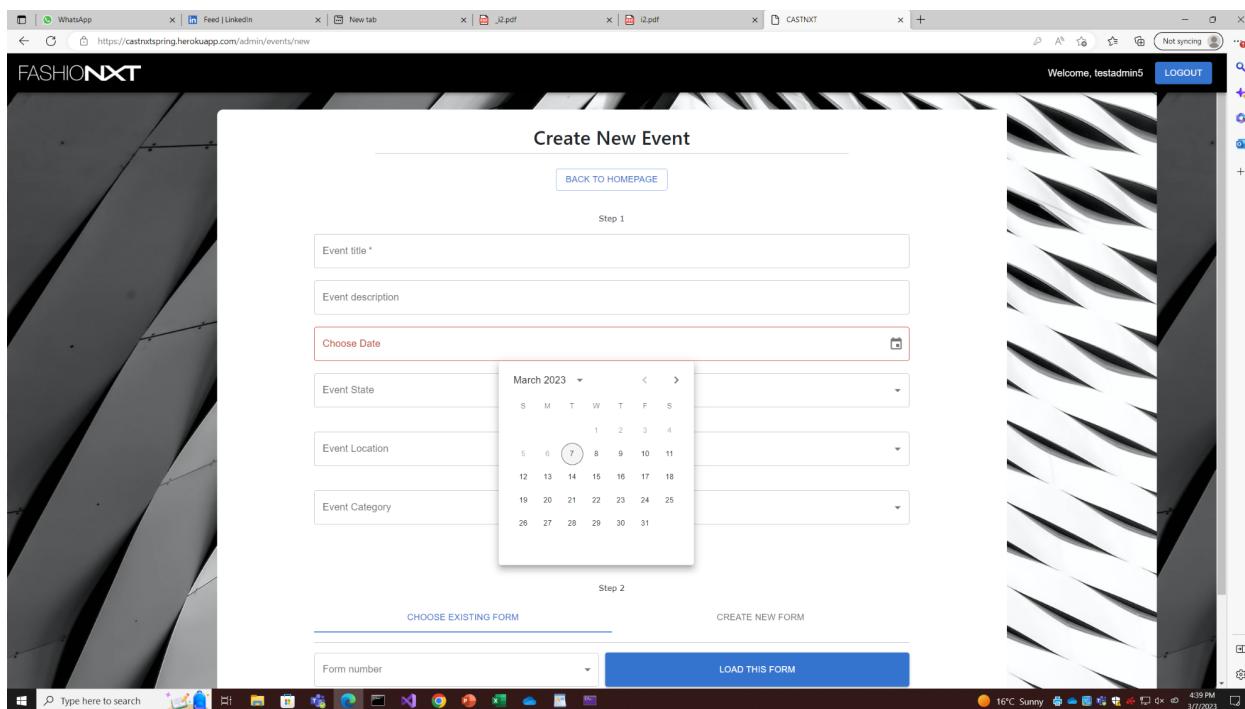
Event	Date	Category	Location	Actions
Xyz				<input type="button" value="Edit"/> <input type="button" value="Delete"/>
Abc				<input type="button" value="Edit"/> <input type="button" value="Delete"/>

### Design Diagrams:

1. Edit Event:



## 2. Adding validations to the fields in UI at multiple locations



FASHIONNXT

Welcome, testadmin5 [LOGOUT](#)

Event description: dasgdgv

Choose Date: 03/09/2023

Event State:

Event Location:

Event Category:

Is this a paid event ?  Yes  No

Step 2

[CHOOSE EXISTING FORM](#) [CREATE NEW FORM](#)

Form number: [LOAD THIS FORM](#)

[CREATE EVENT](#)

Error: Please fill out all required fields

FASHIONNXT

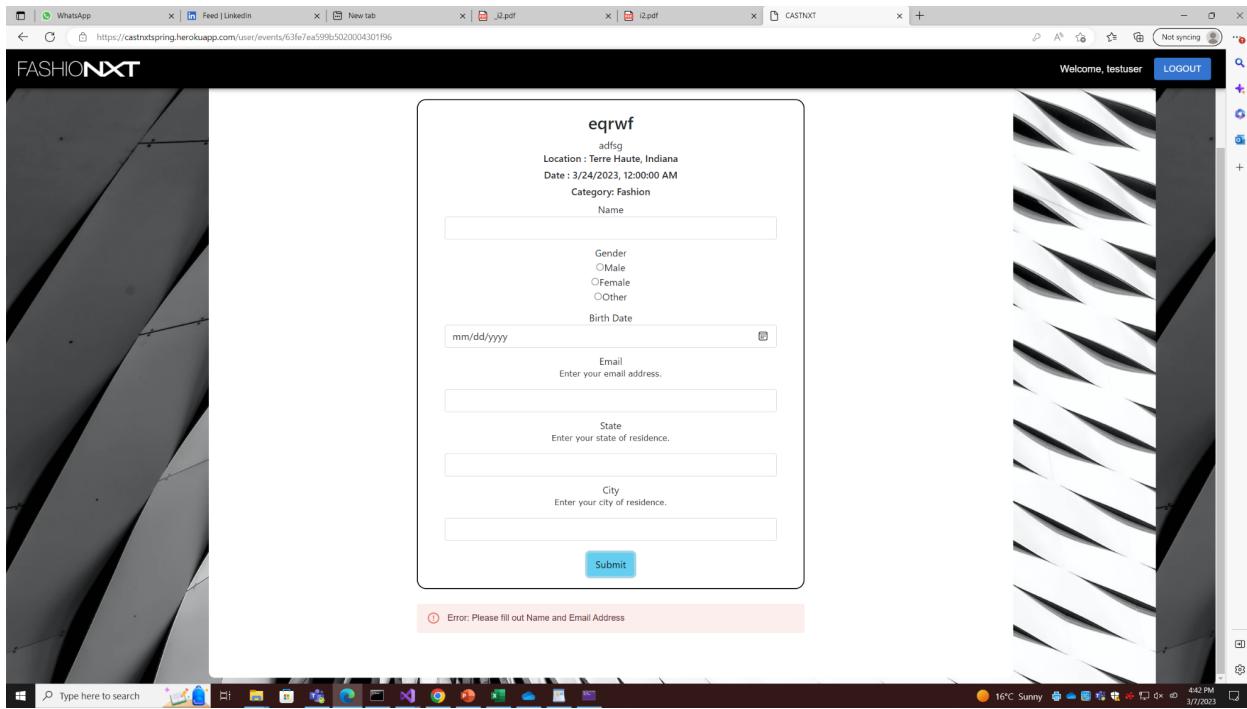
Welcome, testuser [LOGOUT](#)

Event Location:

Is the event paid ? [None](#)

[SUBMIT FILTER](#)

Event	Category	Date	Location	Is
Event	Fashion	2/28/2023	Houston, Texas	
singing	Other	2/28/2023	Honolulu, Hawaii	
testadmin	Fashion	2/28/2023	Washington, District of Columbia	
testadmin1	Performing Arts	2/28/2023	Norwalk, Connecticut	
testadmin2	Music	2/28/2023	Miramar, Florida	
test1	Fashion	2/14/2023	Terre Haute, Indiana	
erwrf	Fashion	3/24/2023	Terre Haute, Indiana	
Event	Fashion	3/15/2023	Brownsville, Texas	
vfs	Performing Arts	3/9/2023	Anderson, Iowa	
ya	Performing Arts	3/9/2023	Sioux City, Iowa	
fbvz	Fashion	3/8/2023	Lexington-Fayette, Kentucky	

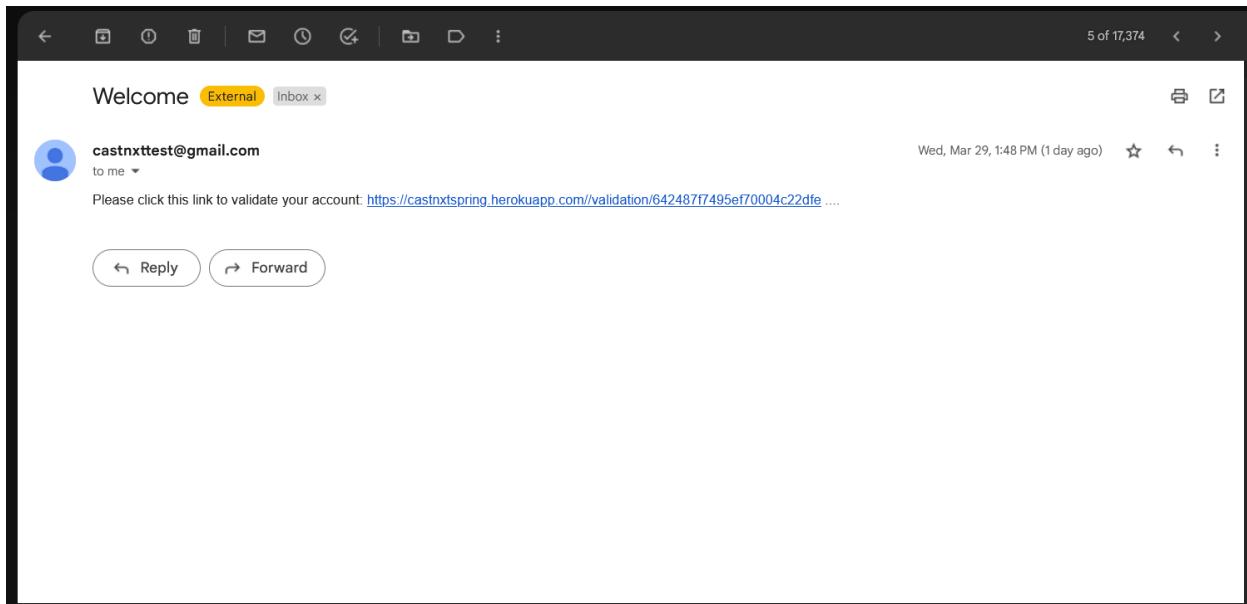


### 3. Fixed issue with webpack hot reload in docker

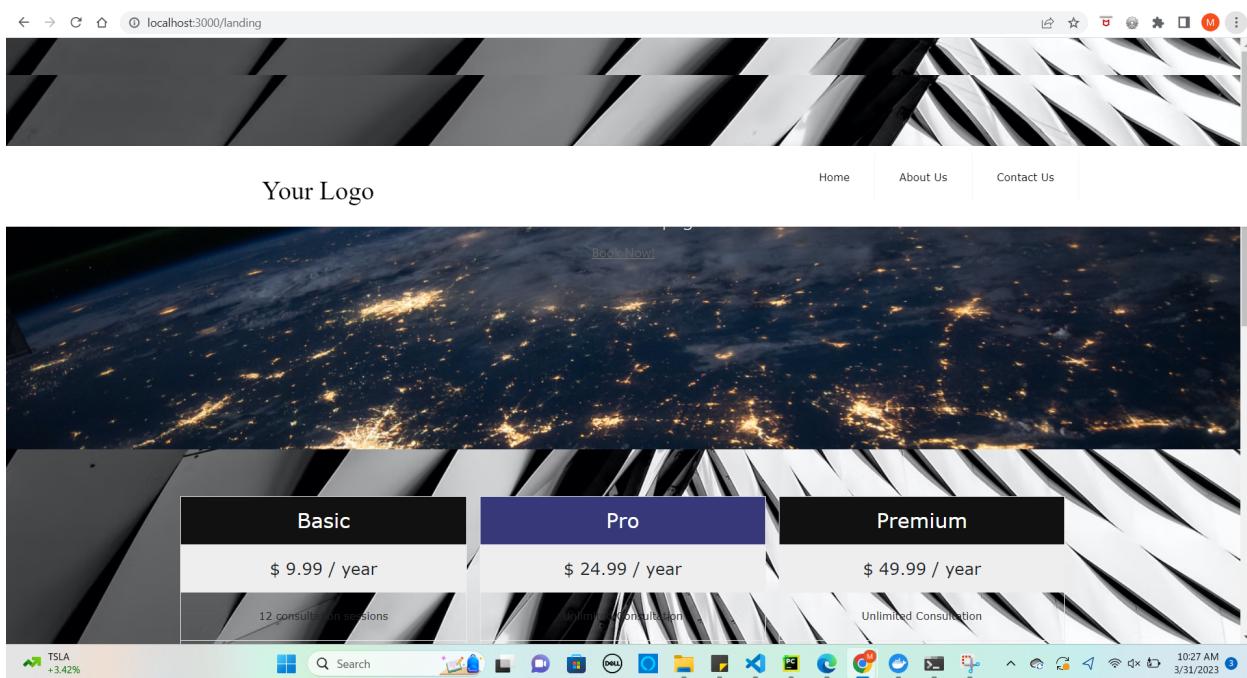
```
C:\Windows\System32\cmd.exe - docker compose up
castnxt-web-1 | 2023-03-06 03:30:56 +0000 Rack app ("GET /packs/application-8947d7238e47d0203a929.js" - (172.21.0.1)): #<Net::ReadTimeout: Net::ReadTimeout>
castnxt-web-1 | 2023-03-06 03:30:56 +0000 Rack app ("GET /packs/hello_react-8947d7238e47d0203a929.js" - (172.21.0.1)): #<Net::ReadTimeout: Net::ReadTimeout>
castnxt-web-1 | [1] wdm| Hash: de3fb782d0a7dfda77b
castnxt-web-1 | Version: webpack 4.46.0
castnxt-web-1 | Time: 151571ms
castnxt-web-1 | Built at: 03/06/2023 3:30:55 AM
castnxt-web-1 |
castnxt-web-1 | Asset           Size     Chunks   Chunk Names
castnxt-web-1 |   Api-de3fb782d0a7dfda77b.js    388 kB  Api [emitted] [immutable] Api
castnxt-web-1 |   Api-de3fb782d0a7dfda77b.js.map 430 kB  Api [emitted] [dev]   Api
castnxt-web-1 |   Main-de3fb782d0a7dfda77b.js    10.5 kB  Main [emitted] [immutable] Main
castnxt-web-1 |   Main-de3fb782d0a7dfda77b.js.map 11.1 kB  Main [emitted] [dev]   Main
castnxt-web-1 |   application-de3fb782d0a7dfda77b.js 402 kB  application [emitted] [immutable] application
castnxt-web-1 |   application-de3fb782d0a7dfda77b.js.map 448 kB  application [emitted] [dev]   application
castnxt-web-1 |   hello_react-de3fb782d0a7dfda77b.js 14.3 kB  hello_react [emitted] [immutable] hello_react
castnxt-web-1 |   hello_react-de3fb782d0a7dfda77b.js.map 11.9 kB  hello_react [emitted] [dev]   hello_react
castnxt-web-1 |   manifest.json      1.42 kB   [emitted]
castnxt-web-1 |   media/assets/images/background-4bb069b7951e3d415e3bbeeb3f15b87.png 355 kB   [emitted]
castnxt-web-1 |   media/assets/images/logo-1675afccca5f5171281faf26afe606e.png 10.3 kB   [emitted]
castnxt-web-1 | [1] wdm| Compiled successfully.
castnxt-web-1 | Started GET "/" for 172.21.0.1 at 2023-03-06 03:31:08 +0000
castnxt-web-1 | Processing by HomeController#index as HTML
castnxt-web-1 |   Rendering home/index.html.erb within layouts/home
castnxt-web-1 |   Rendered home/index.html.erb within layouts/home (0.8ms)
castnxt-web-1 |   Completed 200 OK in 523ms (Views: 463.6ms | MongoDB: 0.0ms)
castnxt-web-1 |
castnxt-db-1 | ["t": {"$date": "2023-03-06T04:28:53.675+00:00"}, "s": "I", "c": "COMMAND", "id": 51883, "ctx": "LogicalSessionCacheReap", "msg": "Slow query", "attr": {"type": "command", "ns": "config.system.sessions", "command": {"listIndexes": "system.sessions", "cursor": {}, "sdb": "config"}, "numYields": 0, "reslen": 245, "locks": {"FeatureCompatibilityVersion": {"acquireCount": {"r": 1}}, "Global": {"acquireCount": {"r": 1}}, "Mutex": {"acquireCount": {"r": 1}}}, "storage": {}, "protocol": "op_msg", "durationMillis": 13151}
castnxt-db-1 | ["t": {"$date": "2023-03-06T04:28:53.675+00:00"}, "s": "I", "c": "INDEX", "id": 5479200, "ctx": "TTLMonitor", "msg": "Deleted expired documents using index", "attr": {"namespace": "config.system.sessions", "index": "lsidTTLIndex", "numDeleted": 0, "durationMillis": 14261}
castnxt-db-1 | ["t": {"$date": "2023-03-06T05:01:17.125+00:00"}, "s": "I", "c": "INDEX", "id": 5479200, "ctx": "TTLMonitor", "msg": "Deleted expired documents using index", "attr": {"namespace": "config.system.sessions", "index": "lsidTTLIndex", "numDeleted": 0, "durationMillis": 1771}]
castnxt-web-1 | Started GET "/" for 172.21.0.1 at 2023-03-06 05:02:07 +0000
castnxt-web-1 | Processing by HomeController#index as HTML
castnxt-web-1 |   Rendering home/index.html.erb within layouts/home
castnxt-web-1 |   Rendered home/index.html.erb within layouts/home (25.7ms)
castnxt-web-1 |   Completed 200 OK in 479ms (Views: 4620.6ms | MongoDB: 0.0ms)
castnxt-web-1 |
castnxt-db-1 | ["t": {"$date": "2023-03-06T05:30:32.580+00:00"}, "s": "I", "c": "COMMAND", "id": 51883, "ctx": "LogicalSessionCacheReap", "msg": "Slow query", "attr": {"type": "command", "ns": "config.system.sessions", "command": {"listIndexes": "system.sessions", "cursor": {}, "sdb": "config"}, "numYields": 0, "reslen": 245, "locks": {"FeatureCompatibilityVersion": {"acquireCount": {"r": 1}}, "Global": {"acquireCount": {"r": 1}}, "Mutex": {"acquireCount": {"r": 1}}}, "storage": {}, "protocol": "op_msg", "durationMillis": 6044}
castnxt-db-1 | ["t": {"$date": "2023-03-06T05:30:32.582+00:00"}, "s": "I", "c": "COMMAND", "id": 51883, "ctx": "LogicalSessionCacheRefresh", "msg": "Slow query", "attr": {"type": "command", "ns": "config.system.sessions", "command": {"listIndexes": "system.sessions", "cursor": {}, "sdb": "config"}, "numYields": 0, "reslen": 245, "locks": {"FeatureCompatibilityVersion": {"acquireCount": {"r": 1}}, "Global": {"acquireCount": {"r": 1}}, "Mutex": {"acquireCount": {"r": 1}}}, "storage": {}, "protocol": "op_msg", "durationMillis": 6033}
castnxt-db-1 | ["t": {"$date": "2023-03-06T05:30:32.584+00:00"}, "s": "I", "c": "COMMAND", "id": 23009, "ctx": "PeriodicTaskRunner", "msg": "Task finished", "attr": {"taskName": "UnusedLockCleaner", "durationMillis": 628}]

72°F Cloudy 7:55 AM 3/8/2023
```

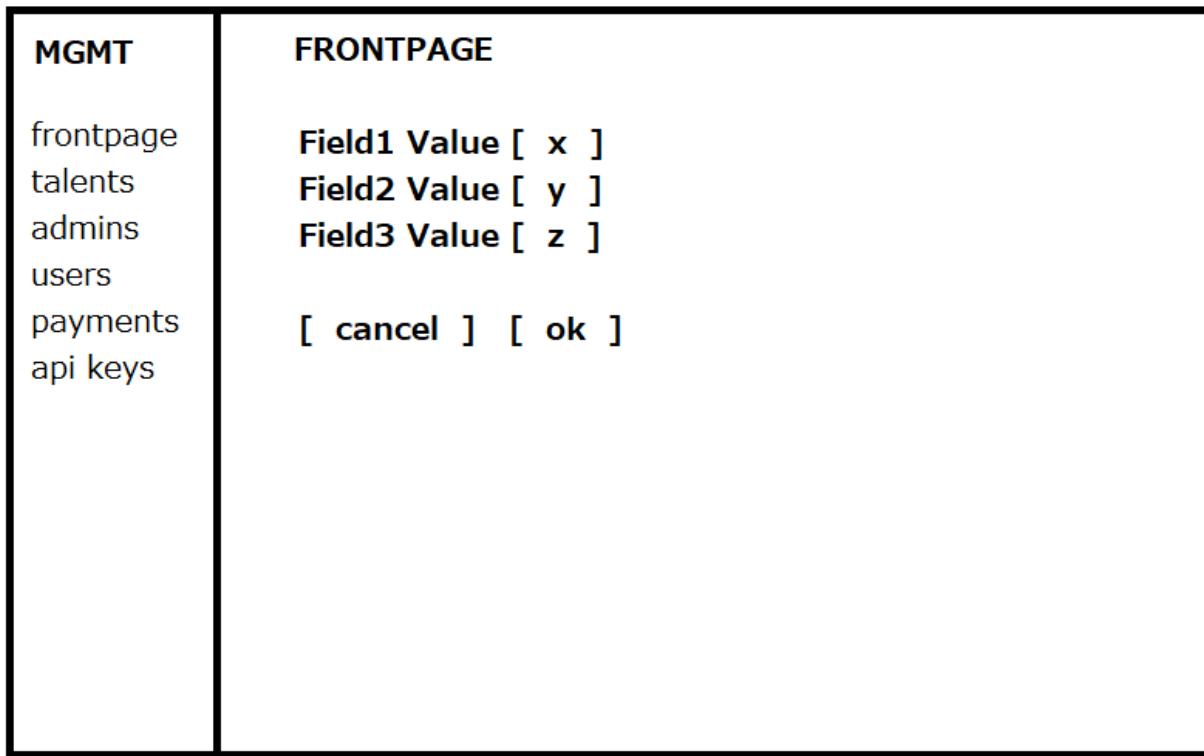
### 4)Email Notification



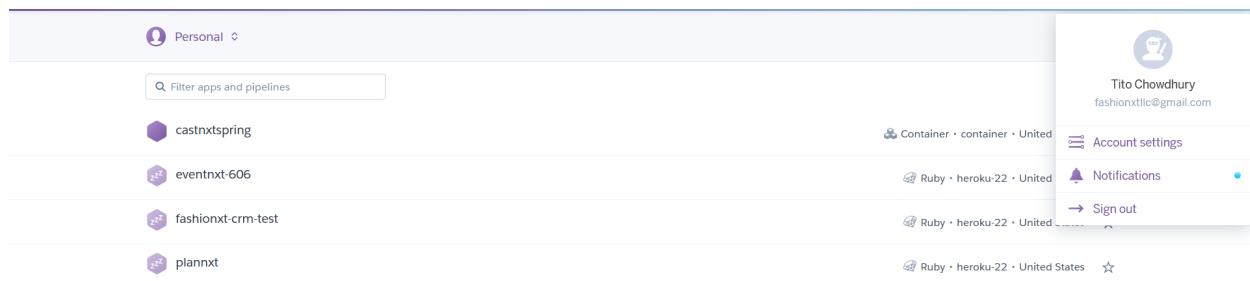
## 5) Landing page



## 6) Site Admin Panel

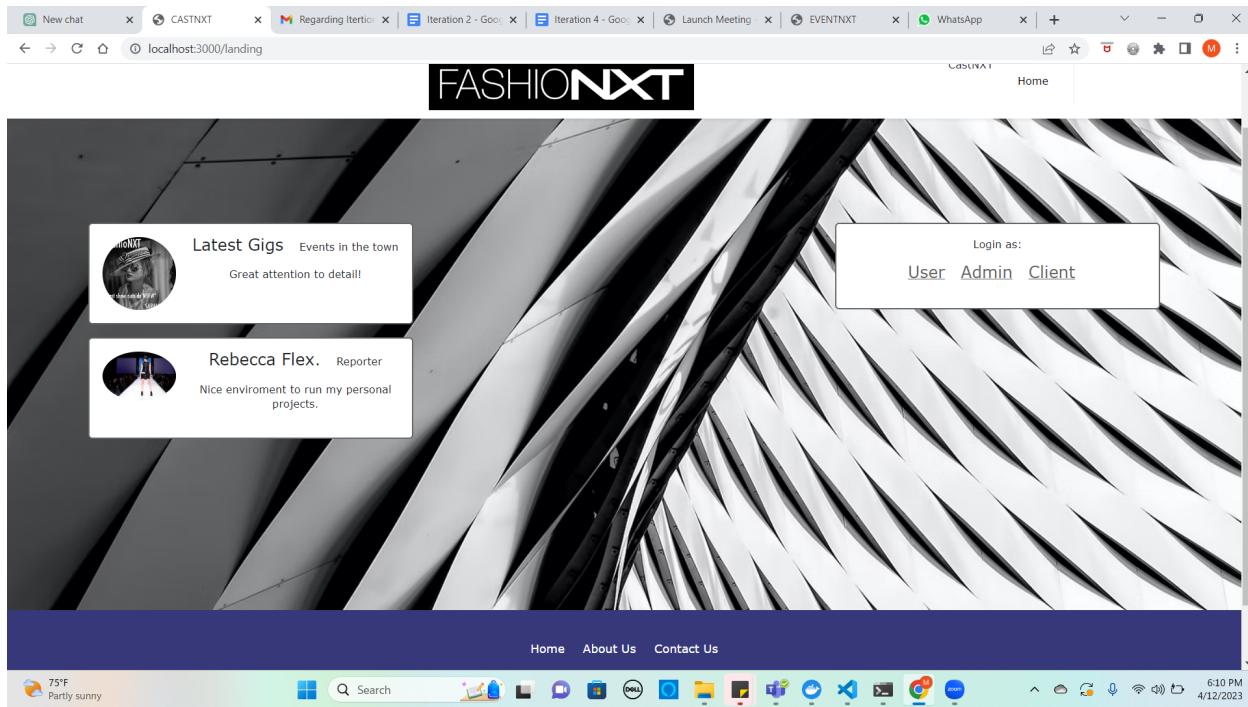


7) Transfer of Heroku and Github accounts for all teams in Events360 to Client

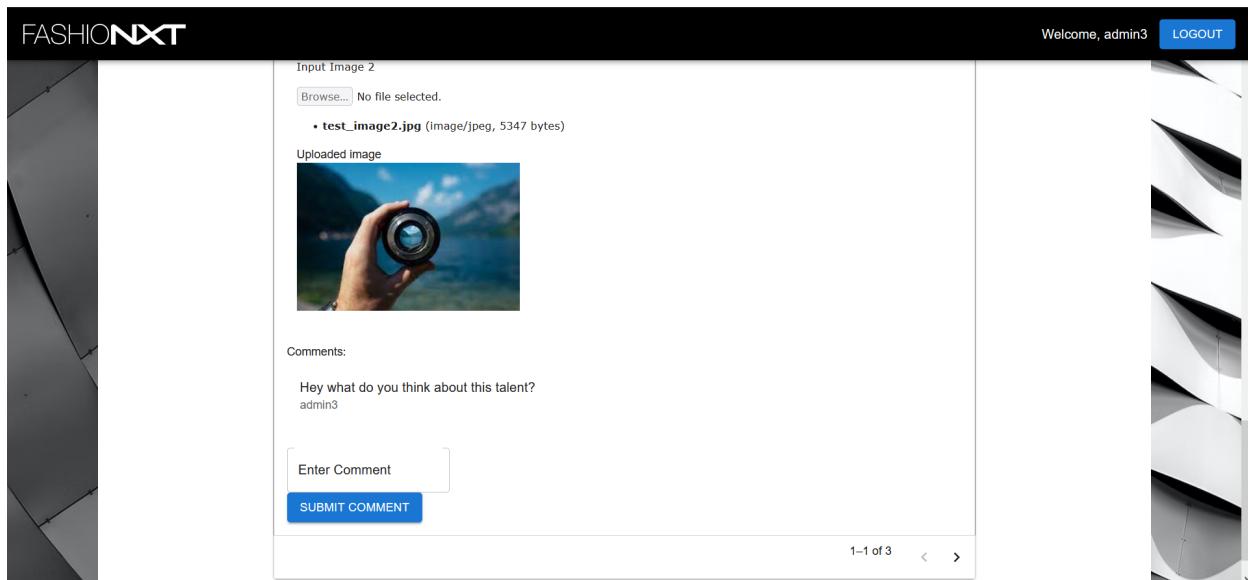


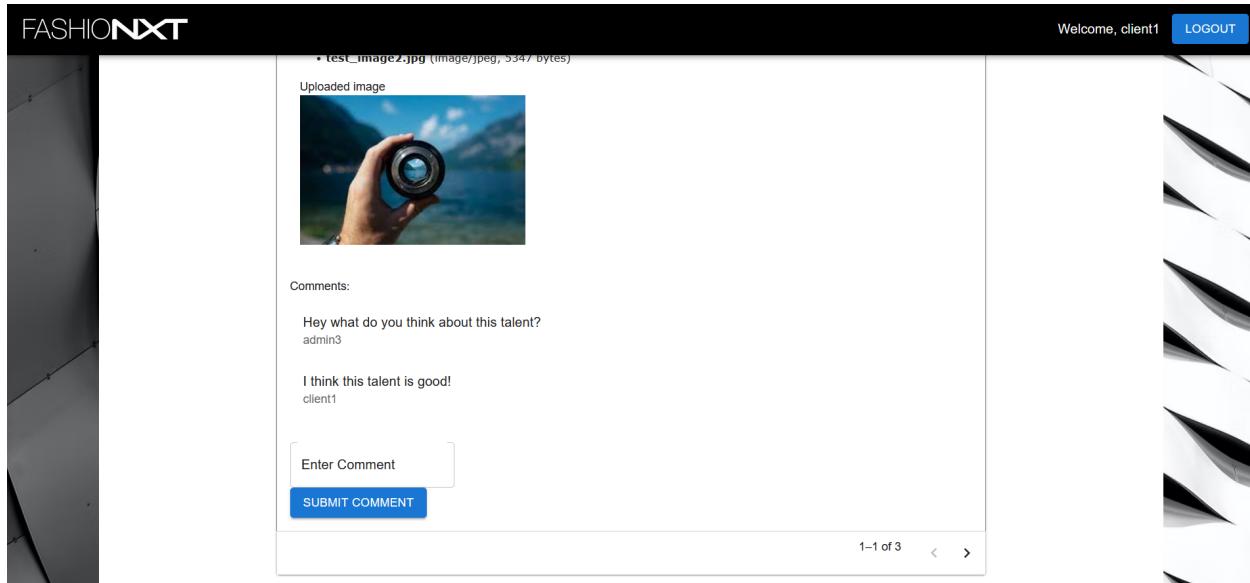
The screenshot shows the Heroku dashboard interface. At the top, there's a navigation bar with a user icon and the text "Personal". Below the navigation bar, there's a search bar labeled "Filter apps and pipelines". A list of four apps is displayed: "castnxtspring", "eventnxt-606", "fashionxt-crm-test", and "plannxt". To the right of the app list, there's a sidebar with the user's profile picture, name "Tito Chowdhury", email "fashionxtlc@gmail.com", and account details: "Container · container · United". The sidebar also includes links for "Account settings", "Notifications", and "Sign out".

## 8) Added user logins, Event details, Navbar, and Footer on the landing page



## 9)1) Chat functionality between the Producer and the Client





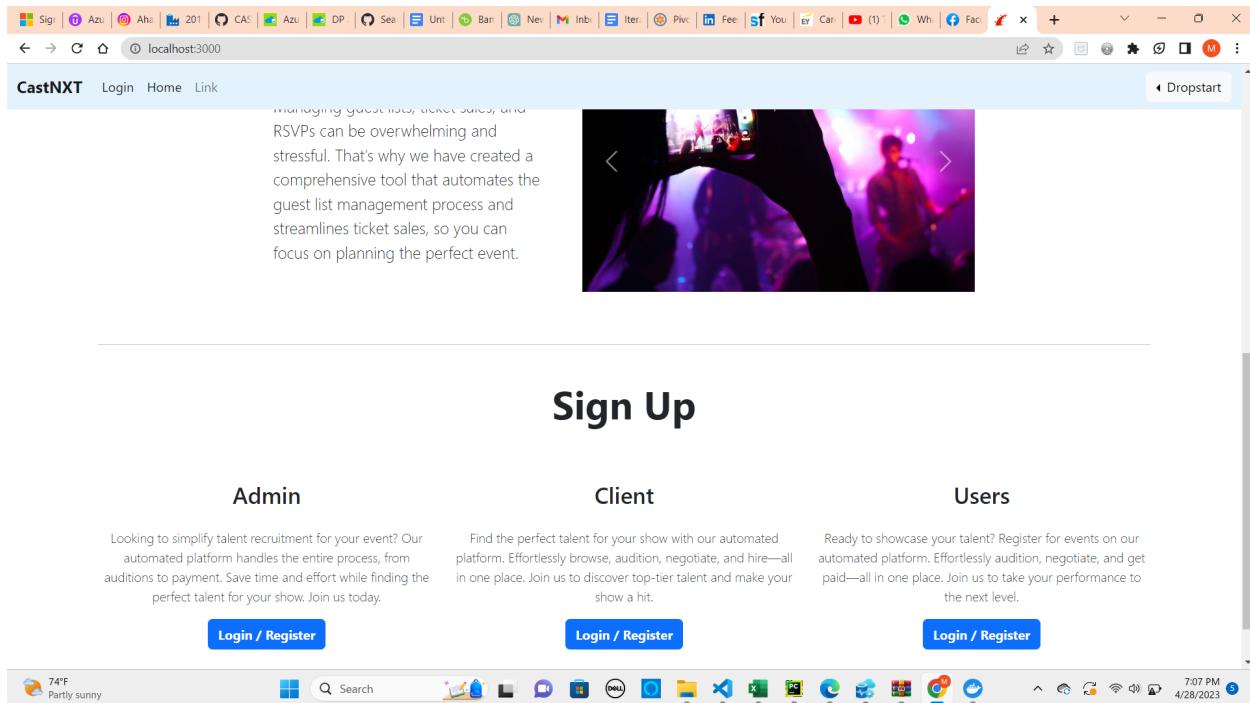
## 11) Landing Page:

CastNXT

Streamline talent recruitment for events and shows with our automated platform. Auditions, placement, replacement, negotiation, handover, and payment—all in one place. Join us for a seamless experience!

At CastNXT, we understand that

74°F Partly sunny



## Improving the Legacy Project:

### Learning

The project has undergone three iterations, which has resulted in a significant amount of information to catch up on. The final report and presentation made by the previous team have been reviewed by all of us. However, we have not been able to explore the application's current state as the Heroku link left by the previous team is not working.

### Improving the code

To enhance the code, our initial focus is to set up the development environment and deploy the project. The client has requested us to spend a week analyzing the application from the perspective of the stakeholders and identify areas for improvement. Since it's very difficult for us to do the setup as the ruby version got changed, we have done containerisation of the code. We have created a docker setup and made improvisations so that anyone can make changes in the UI code without having to restart the docker to see the changes. Also, it will be easy for future team members to run the application directly without facing any issues with the setup. All the steps are documented so that they can just follow the steps and up the application.

### Iteration Accomplishments:

**Iteration 1:** In this iteration, we took the legacy code from the previous sprint and we have created a docker setup for the code and created a deployment setup

Points finished:8

**Iteration 2:** Added UI validations in multiple locations so that users cannot register for backdated events and admin cannot create backdated events. Added edit functionality and fixed web packer hot reload problems

Points finished:15

**Iteration 3:** Added site admin panel for the client so that he can manage the events seamlessly, Fixed ruby credential manager functionality in ruby, Added Email Notification and fixed date picker crushing problems and user registration problems

Points finished: 16

**Iteration 4:** Fixed issues related to events and admin page,

Transferred github and Heroku accounts to all the teams

Added all the contents and navbars, headers in landing page and ported but it got changed because the client wanted altogether a new landing page

Pointsfinsihed: 11

**Iteration 5:** All filters in the application, Unified all the users and payments process, Implemented chat functionality, Added new landing page for the application and ported it as new home page

Pointsfinsihed: 23

## **Testing Report Details:**

### **JEST framework:**

We have added jest framework test cases according to our stories. Below is the report for that test cases. The test coverage is 77% and we have included the same in the report.

### **REACT UI material library:**

For maintaining the design layer consistent across the application we have used the react UI material library. This we have used for the front end of the application.

PASS	spec/javascript/Components/Filter/ <b>IsPaidFilter.spec.js</b>				
PASS	spec/javascript/Components/Admin/ <b>AdminSubmittedDocs.spec.js</b>				
PASS	spec/javascript/Components/Filter/ <b>CategoryFilter.spec.js</b>				
PASS	spec/javascript/Components/Client/ <b>ClientEventSummary.spec.js</b>				
PASS	spec/javascript/Components/User/ <b>UserHomepage.spec.js</b>				
File	% Stmt	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	<b>77.49</b>	<b>62.66</b>	<b>67.36</b>	<b>77.77</b>	
components/Admin	<b>91.3</b>	<b>76</b>	<b>86.48</b>	<b>91.11</b>	
AdminEventHome.js	<b>100</b>	<b>80</b>	<b>100</b>	<b>100</b>	<b>28-45</b>
AdminEventSummary.js	<b>95.55</b>	<b>72.22</b>	<b>100</b>	<b>95.34</b>	<b>28,62</b>
AdminHomepage.js	<b>87.5</b>	<b>83.33</b>	<b>66.66</b>	<b>87.5</b>	<b>41,55</b>
AdminSubmittedDocs.js	<b>95</b>	<b>80</b>	<b>100</b>	<b>94.73</b>	<b>31</b>
AdminUserTable.js	<b>83.33</b>	<b>66.66</b>	<b>72.72</b>	<b>83.33</b>	<b>76-80,84,106</b>
components/Client	<b>68.08</b>	<b>50</b>	<b>66.66</b>	<b>68.47</b>	
ClientEventFeedback.js	<b>58.53</b>	<b>50</b>	<b>50</b>	<b>60</b>	<b>54-58,101-116,123-132,175-198</b>
ClientEventPage.js	<b>100</b>	<b>75</b>	<b>100</b>	<b>100</b>	<b>67</b>
ClientEventSummary.js	<b>65.71</b>	<b>37.5</b>	<b>72.72</b>	<b>64.7</b>	<b>45-56,86-98</b>
ClientHomepage.js	<b>90.9</b>	<b>75</b>	<b>80</b>	<b>90.9</b>	<b>37</b>
components/Filter	<b>66.66</b>	<b>100</b>	<b>50</b>	<b>66.66</b>	
CategoryFilter.js	<b>75</b>	<b>100</b>	<b>50</b>	<b>75</b>	<b>6</b>
IsPaidFilter.js	<b>75</b>	<b>100</b>	<b>50</b>	<b>75</b>	<b>6</b>
LocationFilter.js	<b>61.53</b>	<b>100</b>	<b>50</b>	<b>61.53</b>	<b>17-19,26,40</b>
components/Forms	<b>90.62</b>	<b>90</b>	<b>87.5</b>	<b>90</b>	
FormBuilder.js	<b>90.62</b>	<b>90</b>	<b>87.5</b>	<b>90</b>	<b>53-54,79</b>
components/Navbar	<b>77.77</b>	<b>100</b>	<b>60</b>	<b>75</b>	
Header.js	<b>77.77</b>	<b>100</b>	<b>60</b>	<b>75</b>	<b>26-29</b>
components/Shared	<b>81.48</b>	<b>50</b>	<b>73.33</b>	<b>81.48</b>	
DatePicker.js	<b>77.77</b>	<b>50</b>	<b>60</b>	<b>77.77</b>	<b>22,30,53-54</b>
DatePickerEnd.js	<b>83.33</b>	<b>50</b>	<b>80</b>	<b>83.33</b>	<b>21,29,52</b>
DatePickerStart.js	<b>83.33</b>	<b>50</b>	<b>80</b>	<b>83.33</b>	<b>22,30,53</b>
components/User	<b>56.57</b>	<b>50</b>	<b>30.43</b>	<b>58.1</b>	
UserHomepage.js	<b>56.57</b>	<b>50</b>	<b>30.43</b>	<b>58.1</b>	<b>...108,112-115,119-122,135,141,165,202-214,239</b>
utils	<b>80.85</b>	<b>64.28</b>	<b>80</b>	<b>82.22</b>	
AdminDataUtils.js	<b>100</b>	<b>80</b>	<b>100</b>	<b>100</b>	<b>3-12</b>
DataParser.js	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	
FormsUtils.js	<b>61.9</b>	<b>25</b>	<b>66.66</b>	<b>61.9</b>	<b>1125,1177-1188</b>
RangeFilter.js	<b>85.71</b>	<b>100</b>	<b>50</b>	<b>100</b>	

```

Test Suites: 17 passed, 17 total
Tests:       36 passed, 36 total
Snapshots:   26 passed, 26 total
Time:        4.478 s

```

## Team Roles:

The team roles have changed over the course of the project as per the instructor's advice in the beginning. For each iteration, there was a different scrum master and a different product owner. All of us, now, have the experience of being a scrum master and a product owner. Below are the roles for each iteration we had:

Iteration	Scrum Master	Product Owner

O	Anshul Mendiretta	Manaswini Devabhakthuni
---	-------------------	-------------------------

1	Manaswini Devabhakthuni	Anshul Mendiratta
2	Kelvin Wang	Manaswini Devabhakthuni
3	Kelvin Wang	Anshul Mendiretta
4	Anshul Mendiretta	Kelvin Wang
5	Amalesh Arivanan	Ming Li

## Meetings with the client:

Throughout the course of the semester, we met the client every week on Wednesdays (5 PM - 6 PM) on the Zoom platform. The general format of the meeting -

1. The product owner will discuss the work that has been completed during the week.
2. Each developer will demonstrate their individual work during the meeting.
3. The client will be asked to provide feedback on the features that have been developed so far.
4. The client will provide suggestions for the next week's work and prioritize the features to be developed.

### Meeting 1:

Date: February 7, 2023

#### Minutes of the Meeting:

1. Team introductions with the client

2. The client has given an overview of what the project is about.
3. Discussed the legacy changes of the app over the course of the last 2 semesters
4. Discussed the high-level requirements.

Meeting 2:

Date: February 17th 2023

Minutes of the Meeting:

- 1)We were having trouble getting the application up and running locally and on Heroku. Emphasis was placed on getting it running as soon as possible.
- 2)Adding a payment method and upgrading Ruby to newer versions were discussed as potential user stories going forward.

Meeting 3:

Date : February 24th 2023

Minutes of the meeting:

1. We explained how much we have understood the previous team's work and code.
2. We proposed some feature enhancements and discussed their scope and priority. Narrowed down to some important features as per client's requirements.
3. Discussed payment feature, Edit feature with the client and started working on that
4. Deprioritize forgot password feature as the ca client has a plan to implement a single sign-on

Meeting 4:

Date: March 1st 2023

Minutes of the meeting:

- 1)In the first meeting, the needs of a Cast NXT and Events 360 collaboration were explored.
- 2)A future SAML implementation was set into motion but unfortunately could not progress due to a lack of response from the other team.
- 3)We discussed the implementation of payment features.

### Meeting 5:

Date: March 8th 2023

Minutes of the meeting:

1. We discussed the improvements to the existing tests in the project from the previous team.
2. We discussed fixes to critical bugs in the application left by the previous team.
3. We resolved issues with sendblue and received login credentials to monitor the accounts.

### Meeting 6:

Date: March 22, 2023,

Minutes of the meeting:

- 1)Discussed the implementation of the Landing page and how to implement SAML and Landing page
- 2)Discussed enabling multiple roles to the user to be user/admin/client
- 3)Discussed enhancements on payment activities

### Meeting 7:

Date: March 29th 2023

Minutes of the meeting:

- 1)Discussed on demo and fixing the bugs
- 2)Discussed on improving the test cases.

### Meeting 8:

Date: April 5th 2023

Minutes of the meeting:

- 1)customer showed an example of a landing page style he desired
- 2)customer asked about certain changes and if they could be implemented for heroku / CICD
- 3)customer wanted to check on certain features in the live app but couldn't due to certain bugs

### Meeting 9:

Date: April 12th 2023

Manaswini - landing page modifications

Anshul - Github/heroku change proposals

Ming - Functionality improvements and fixes

Amalesh - Restrictions on implementation of a complete payment system

Kelvin - Merging of 3 roles into 1 user, messaging, SAML

Meeting 10:

Date: April 19th 2023

- 1)The client discussed on the major deliverables that are still in progress

Meeting 11:

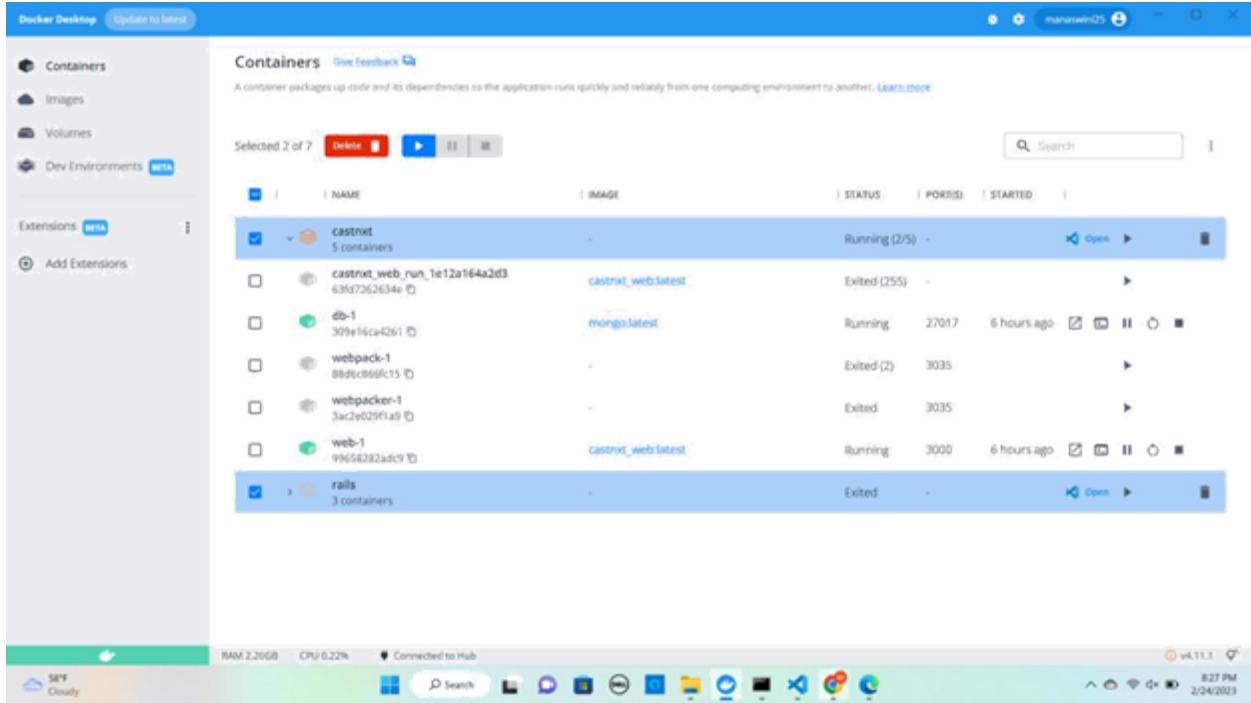
Date: April 26th 2023

- 1)Demonstration of App to the client

Run Application on Docker Steps:

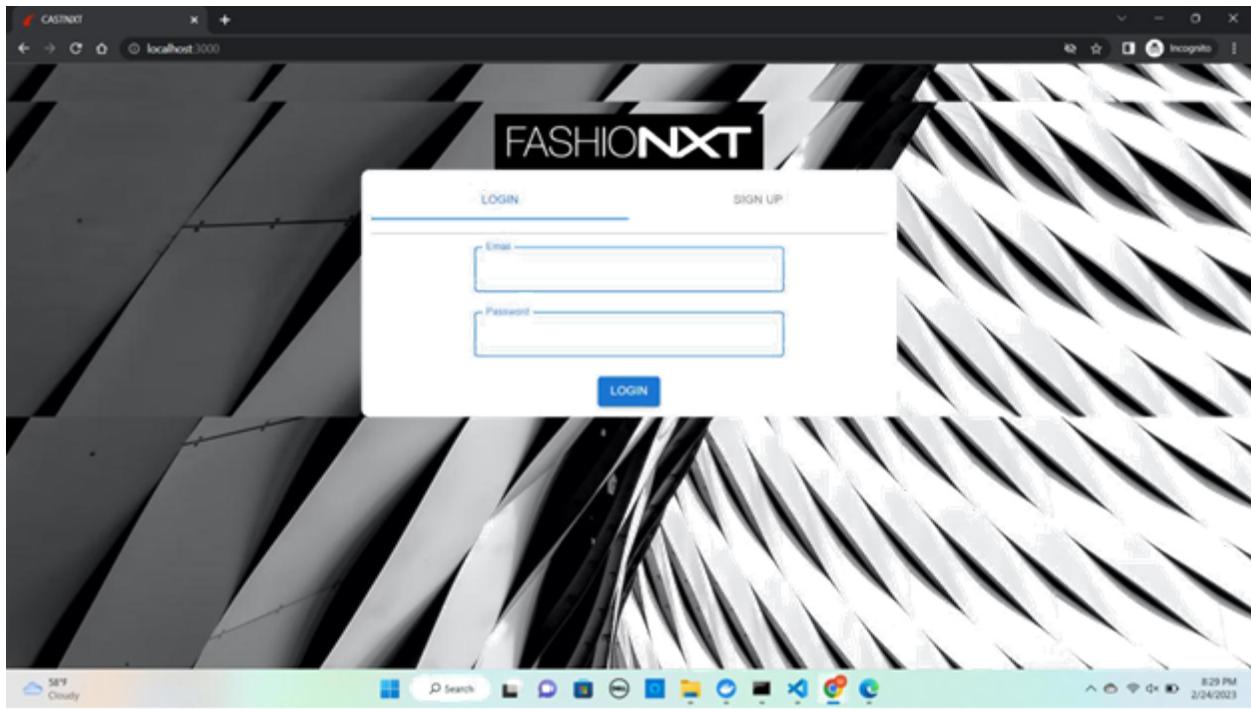
- 1)Install docker desktop
- 2)Start docker desktop
- 3)Navigate to folder web/CastNXT in cmd or terminal
- 4)Run the command "docker build" -- optional but good to run
- 5)run the command "docker-compose up"
- 5)Open localhost:3000 in browser

- 1)Docker container is created with database and web pack



2)

### 3) Launched Application using docker:



### Heroku deployment steps:

Clone the repository:

Connect to Heroku and Create a Deployment

```
heroku login -i
```

```
heroku container: login heroku create -a <app>
```

Push code into the Deployment

```
heroku container:push web -a <app name> | Heroku release web -a <app name>
```

Create a Free MongoDB Atlas Cluster

Configure the Free MongoDB Atlas Cluster under Security Quickstart

Get MongoDB Atlas Connection URL: Make sure to add the database name in the URL while configuring the Mongo DB URL.

Connect Deployment to MongoDB Atlas on Heroku.com

Issues while doing Heroku deployment:

1. We faced connection issues from the Heroku app to the MongoDB cluster. The database name is not included in the connection string by default.
2. From Heroku settings we have to add config variables for example MONGODB\_URI. In our case, the connection string is :

```
mongodb+srv://castnxt:<password>@cluster0.9mfb0kl.mongodb.net/castnxttest?  
retryWrites=true&w=majority
```

3. To resolve this, the database name and password must be given in the connection string.

### **Repo contents and deployment scripts:**

1. Heroku deployment is being done manually using Heroku CLI. The commands are given in the Deployment Guide section.
2. Mongdb.yml files have all the database configurations for production, development and test environments.
3. We have created a single GitHub repository and all the code is pushed to main branch. Few people have their own branches but finally merged to main branch.

### **Github Repo details:**

File Edit Selection View Go Run Terminal Help

EXPLORER

CASTNXT

- app
  - assets
  - config
  - images
  - javascripts
  - stylesheets
- channels\application\_cable
  - channel.rb
  - connection.rb
- controllers
  - concerns
  - admin\_controller.rb
  - api\_controller.rb
  - application\_controller.rb
  - client\_controller.rb
  - events\_controller.rb
  - forms\_controller.rb
  - home\_controller.rb
  - negotiations\_controller.rb
  - slides\_controller.rb
  - user\_controller.rb
  - user\_mailer.rb
- helpers
  - admin\_helper.rb
  - application\_helper.rb
  - client\_helper.rb
  - events\_helper.rb
  - forms\_helper.rb
  - home\_helper.rb

...

JS application  
app > view

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37

aws

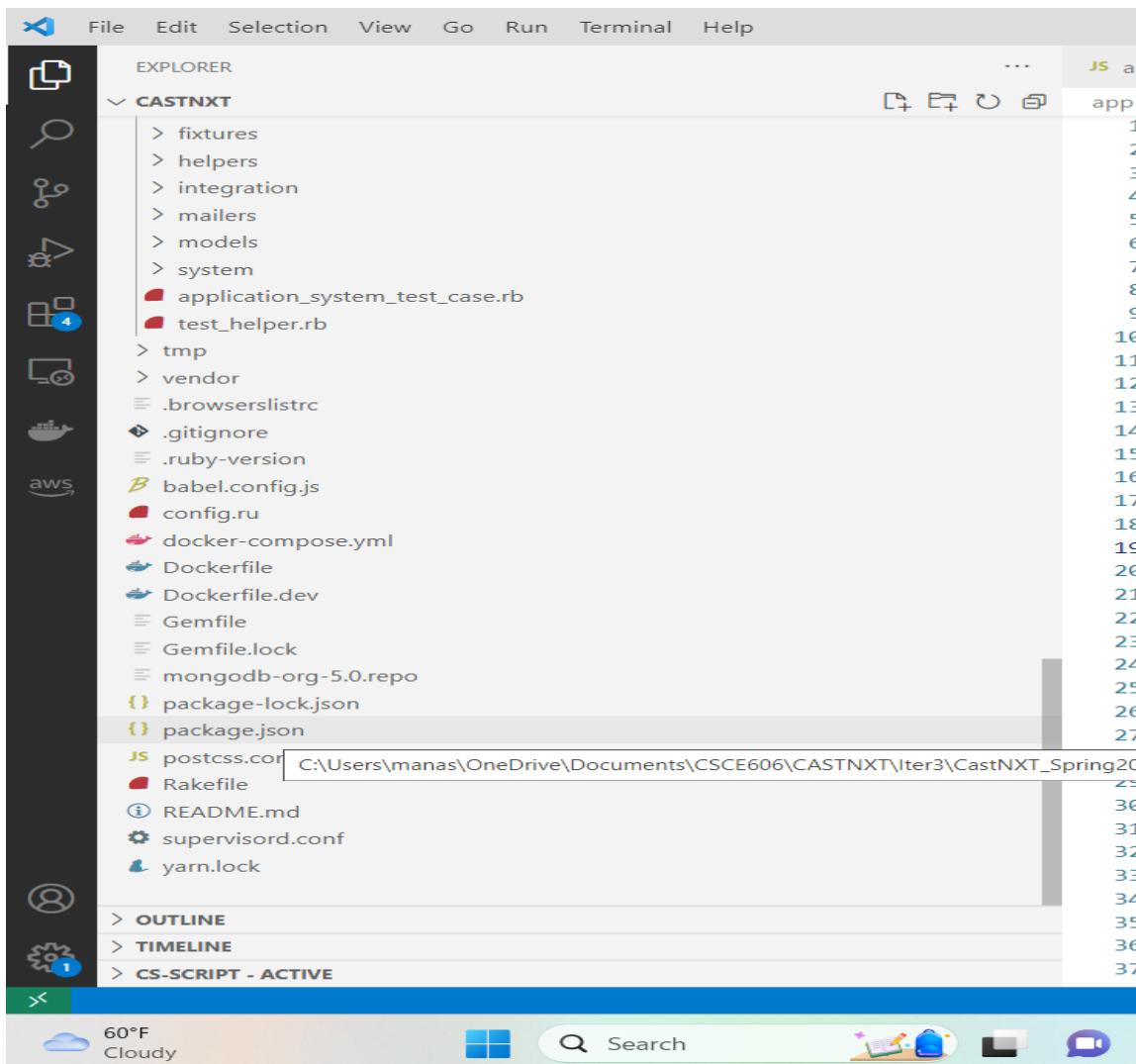
OUTLINE

TIMELINE

CS-SCRIPT - ACTIVE

60°F Cloudy

Search



## Pending Items and Future Outlook:

1. To enable producers and clients to easily identify and categorize talents for future reference, a colour coding feature should be available on the slide tab. This feature will allow them to assign different colours to talents based on their types or categories. This way, they can quickly and easily locate specific talents when they need them in the future.
2. When talent is removed from the admin deck, which is the first level of filtering submissions, it should also be automatically removed from the assigned client decks that are being negotiated on the user interface. This bug needs to be fixed to ensure that the talent list remains updated and consistent across all decks.

3. Provide support for previewing PDF uploads. Currently, the application supports previewing only image uploads
4. Mobile View: Optimize the application for viewing on mobile devices.
5. Viewing content in the form of tiles not as grid

#### **Important Links:**

- 1)Pivotal Tracker - <https://www.pivotaltracker.com/n/projects/2630374>
- 2)SlackChannel - <https://app.slack.com/client/T03GDG6CE/C04NJA2V39D>
- 3) GitHub - [https://github.com/FashioNXT/CastNXT\\_Spring2023](https://github.com/FashioNXT/CastNXT_Spring2023)
- 4)Heroku - <https://castnxtspring.herokuapp.com/>
- 5)Presentation Demo URL:  
<https://docs.google.com/presentation/d/1S0nHeh9L5jiQ8s7EH45dPrWglNu3xM9pw3FT8IkAPZQ/edit#slide=id.p>