

Things to do! NY

Development Document

UI Eagles:

Meril Dsouza (msd2178)

Yumeng Liao (yl2908)

Sahana Subbanna (ss4737)

Kira Ullman (khu2002)

1. Overall Process

When we first read through the description of the project, we were really excited and ready to make the best use of what the NY Times API had to offer. Having used the Community API before, we knew NY Times API could give us many more such APIs with a wealth of information. When we browsed through the available APIs, we felt the **Event Listing API** would be a really good API to use since it gave us information about the category of the event, location, if it was free etc. Hence we came up with the idea of an app that would list event details from the NY Times API in an aggregated manner letting the user go through multiple event listings in an area and viewing the location for it.

We decided we wanted our app to cater working individuals in the age group 20-30 since they have money and the zeal to try out new events.

We created an initial prototype to test the functionality of the Event Listing API with basic search parameters of date and radius and integrated the Google Maps API for displaying the map. We chose the center of map to be Columbia University and did a search around the campus. We used myBalsamiq to create our initial prototype before writing the code.

However, later we realised it would be more intuitive to have the events centered around the 5 boroughs of NYC since the API returned events throughout NYC. This led to a search criteria for the borough along with the date and radius. We also felt that we needed to provide the user with the ability to search for specific types of events such as Movies, Theater, Jazz etc. because searching for all events may lead to an overload of superfluous information.

We went with a card-based UI since it was visually appealing for information returned in a list-based format and could neatly contain all relevant information for one particular event in a single block. Also, events are easily separated from each other, making it clear that buttons would apply to one particular event. We added next/prev buttons for browsing through the pages of results, which are only visible when necessary to avoid cluttering the interface.

After internal testing, we realized that a bookmark feature which persisted across sessions would be very helpful for a user who would like to get back to deciding on the event at a later time, possibly with collaboration from his associates.

We wanted to create a responsive design for the app to make the app more intuitive and easy to use since the user gets immediate feedback for his actions. We have used the modal feature of Bootstrap for displaying pop up windows which would disappear when clicked on any area

on the page and also provide smooth transitions between pop-up and pop-out. The pins on the map emerge in a drop-animation, showing the user that their queries have yielded results.

We then tested with external users, who gave us valuable feedback that allowed us to arrive at our final design.

Application Description

“Things to do! NY” is an application which can be used by users to make plans for certain events they wish to attend. This application makes use of the **New York Times’ Event Listing API** to retrieve the relevant information and present to the users.

A user can first choose to filter the sort of events he would like to view. This filtering can be performed based on event date, search radius of the event from a particular borough, and the event type. If the user does not wish to filter events, then default values will be used for the search. The system will then display all retrieved events from which the user can make a selection. The events are displayed 20 at a time and next/prev buttons are provided to page through the results.

Each event will be listed along with its name, a brief description, the event type, and the date-time description if available. There also exists a map on the page with markers denoting the location of each of the event so that a user can also view the directions to the destination, since the exact location may also be a deciding factor when it comes to planning an event in the case of some users.

A user has the ability to bookmark events for future reference and look back at all the bookmarked events at a later stage by clicking on the **My Bookmarked Events** option at the menu at the top right of the page. The list of bookmarked events can be accessed at any time and it is also displayed each time an event is bookmarked to allow for a responsive and informative design. The user can delete individual bookmarks or all the bookmarks on the list. Users are also informed about the **New York Times Critics’ Pick events** which is denoted by a star and the phrase “NY Times Critics’ Pick” next to the corresponding event.

The **Event Page** button opens up a new tab which takes the user to the event in the New York Times website. This button is present on each of the event cards displayed on the page as well as in the window which pops up when a user clicks on the corresponding marker on the map.

The **Home** button clears up the previous results and takes the user back to the start page with the default search fields while retaining the list of bookmarked events from the previous search.

The **About** button provides a brief description of the applications.

The **Help** button provides the user with some basic steps on how to use the application.

Roles & Responsibilities:

Meril: Worked on aspects of the User Interface. Worked with the Google Maps API for the map displayed on the screen which marks the locations of the events retrieved from the search. Worked on the Development Document, User Manual, and making of the video.

Yumeng: Worked on all iterations of front end HTML and CSS using bootstrap. Worked on JS for calendar widget, next and previous buttons, and bookmarking functionality. Performed testing and analysis for both use scenarios. Made Balsamiq designs and worked on the Development Document.

Sahana: Worked on all the iterations of the JS and CSS styling for the card-based UI adding HTML dynamically for the cards inside JS. Worked on querying the NYT API and the Google Maps API including display of the maps, markers, and infowindows. Was involved in the development document including reviewing the heuristic analysis and user manual.

Kira: Worked on all iterations of the back-end and NYTimes API usage. Collaborated on development document and wrote user manual. Performed testing and analysis for both use scenarios. Made Balsamiq designs, pre-testing storyboard, and final storyboard. Worked on final video as well.

2. Target Users

User Class:

Our user class consists of young working professionals, aged 20-30. These people are busy, but also interested in going out, meeting new people, and trying new things. They are adventurous and full of life, albeit pressed for time and always on the go.

Personas:

Persona 1: Greggory (Gregg) Granger, age 22, just graduated from Columbia. He's working at Goldman Sachs as an analyst. He has time only in the evenings, but has plenty of spending money. He spends a lot of time poring over information, and so he likes to find different events to help him decompress. Gregg is not dating anyone, but has his eye on a woman in his office.

Persona 2: Stephanie Lovegood, 28, loves French films, opera, and Oprah. She owns a flower shop called Flowerpuff. She doesn't mind the numbers side of the work, but her passion is making beautiful arrangements. She looks for beauty everywhere she goes. Stephanie works seven days a week, but has flexible hours. She dreams of one day moving Flowerpuff to Paris.

Use Scenarios:

Use Scenario 1: Gregg just asked a woman from the office on a date, and is looking for an event to take his new lady to. It should be classy and fun, but it has to be this Friday night. He would also like for it to be somewhere on the Lower East Side so that they can walk to the Brooklyn Bridge afterwards. He should search for events in Manhattan on Friday night, and then bookmark any events that look interesting and are close to the bridge. He should also navigate to the event pages of each of the events he bookmarked. Finally, he should look on the map to see which event is closest to the Brooklyn Bridge. Once he finds that event, he should navigate to that event's page to check out more about the event and see if he can tell how much the event costs.

Use Scenario 2: Stephanie has recently read a bunch of books on Paris that she really likes, and so she would like to see a French movie. She doesn't need to be up early tomorrow, and she would love to discover a new part of town. She should search for movies with the maximum search radius since she is adventurous. She would like to see a movie in French or about France, and ideally it will be a critically-renowned film - a NYTimes Critics' Pick! Stephanie bookmarks a few movies that she would like to see. Then, she will close the window, and come back later to read more about her bookmarked links and finalize her decision.

3. Design Decisions

In our design, we were careful to adhere to Nielsen's 10 usability heuristics in order to optimize the user interface. Here are the ways we did so:

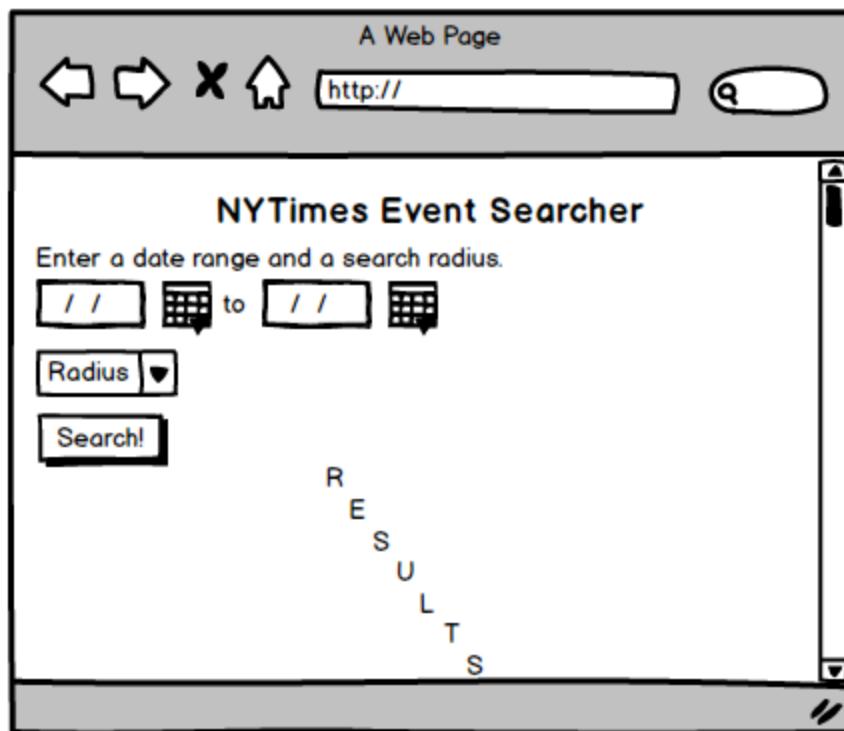
- 1) **Visibility of system status:** The system provides the user with quick and responsive feedback. The color of a button changes when a user hovers the mouse over it. The markers on the map display the name of the event when the mouse is hovered over it. When a user bookmarks an event, the My Bookmarked Events list pops up to show the new addition to the list.
- 2) **Match between system and the real world** - We use commonly-seen language and features. For example, we use the word "search" and "radius," both terms often associated with looking for things near you. We also added a map by using the Google Maps API, which will be a familiar format to most users.
- 3) **User control and freedom** - We have kept the search options at the top of the page, so that if a user makes a mistake in their search, he/she can easily perform the search again with the correct filters. This allows the user to refine his search without too much trouble. Each window has a close button allowing the user to close the windows when they open them, either deliberately or by mistake.
- 4) **Consistency and standards** -We have maintained the common standard of having the search and menu bars at the top of the page making the page familiar to the user who would have used any modern day website. The name of the app is visible at all times to the user. The cards are annotated with the map marker to relate it to the markers seen on the map. Every search returns the results in the same format. This consistency makes our application easy to use and easy to read. The user has an option to page through the results and go back to previous pages.
- 5) **Error prevention** - If the user makes a mistake in their search fields, we have alerts set up to notify him/her. For example, if the user puts the start date later than the end date, the application will not perform the search and will instead alert the user to his/her mistake. The next button is not visible if there are no more than 20 results or if the user is on the last page of the results. This prevents the user from clicking on the next button and waiting for something to happen or wondering why nothing is happening. For the same reasons, the prev button is not visible when the user is on the first page.

- 6) **Recognition rather than recall** - We leave our search criteria as well as all of the results on the page so that users do not have to remember what they searched for or what results they already saw. We also label things clearly so that users do not have to remember, for examples, what buttons do what. There are information windows that pop up when a map marker is clicked on the map which displays the name of the event and the link to the event so that the user can go to the event page from the map marker itself.
- 7) **Flexibility and efficiency of use** - Our design is streamlined and fast, so experienced users need not click through pages of information that they do not need. They simply enter their search criteria and they're off! We also store bookmarked events so that experienced users can navigate directly to events that they have already tagged.
- 8) **Aesthetic and minimalist design** - Our design intentionally minimizes unneeded text or distractions. The color scheme is simple, so as not to distract the user, and we provide some relevant text (while providing the option for the user to find more information about any given event).
- 9) **Help users recognize, diagnose, and recover from errors** - Alerts as well as our help window allow users to self-diagnose common problems that they might come across.
- 10) **Help and documentation** - We have a help window that the users can select if they get stuck. We also have an “About” feature that talks about the structure of the app overall. We have created a user manual with picture annotations to help the uninitiated.

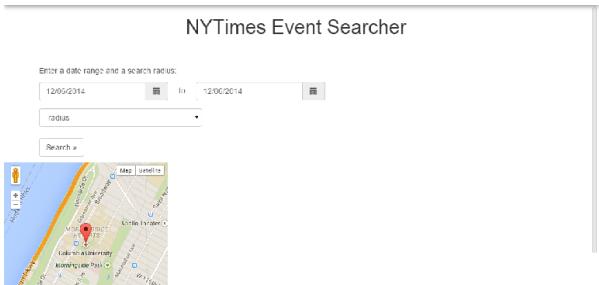
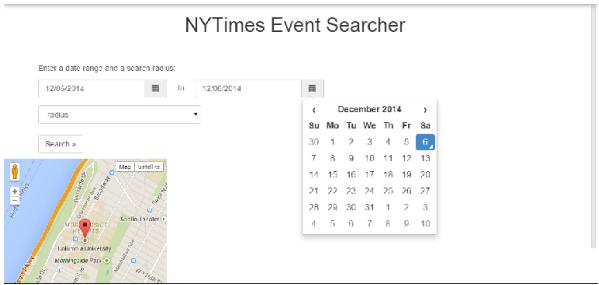
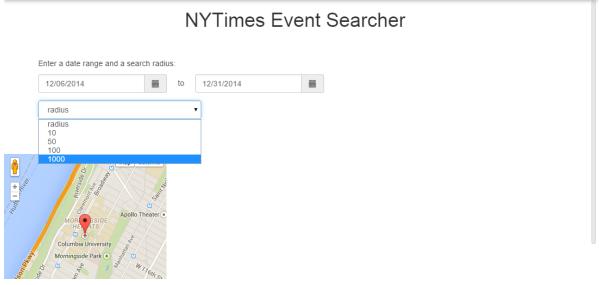
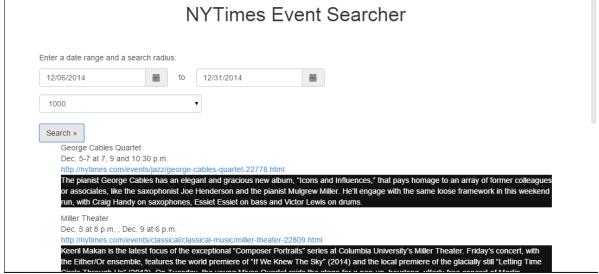
4. Prototyping and Testing

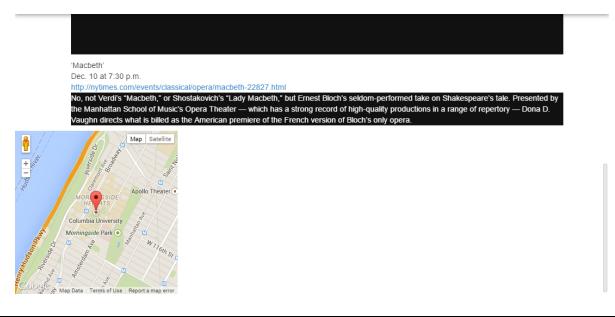
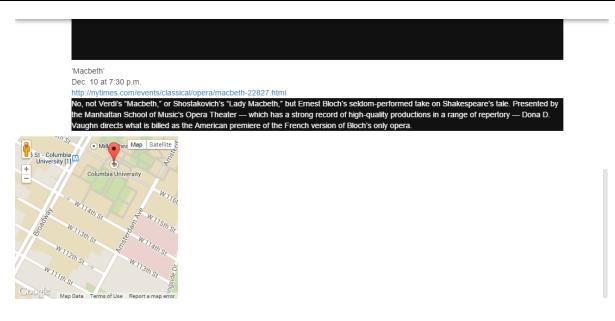
Initial Design:

In our initial design meeting, we decided to have two main search criteria, radius and date range. The radius we decided to center at Columbia University, since we are all here. Both of these fields are directly searchable in the ajax call to the New York Times API, which also factored into our decision to use those fields specifically. We designed a basic balsamiq prototype to guide our development (screenshot below). From this template, we began to test API calls and build our html.



Pre-Testing Storyboard:

	<p>Landing Page</p>
	<p>Search Page</p>
	<p>The default start date and end date are both today's date. User selects a new end date.</p>
	<p>User selects a search radius.</p>
	<p>User clicks "Search" button, and the results are displayed below.</p>

	<p>User scrolls to the bottom of the results page.</p>
	<p>User uses Google Maps controls to navigate in the map.</p>

Testing (performed on tentative version of final UI):

First Set of Tests:

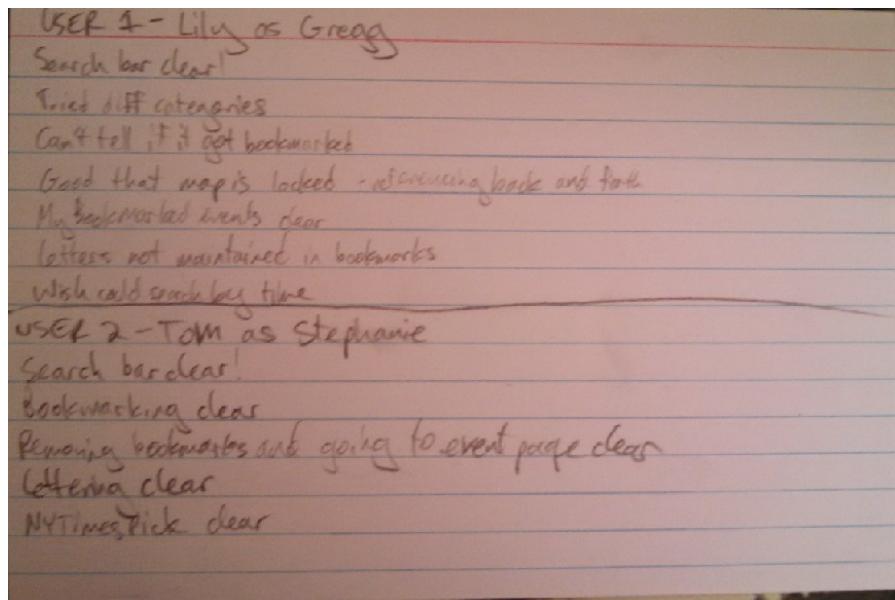
First Test User: Lily as Greggory Granger

Lily was able to navigate the search bar easily, and performed multiple searches with different categories. She wanted to mark down events that she liked, and had trouble remembering what events she had been interested in. Lily kept going back and forth between the results and the map, so locking the map (so that it is always visible) would be great for her. She wished that she could search by time.

Second Test User: Tom as Stephanie Lovegood

Tom navigated the search bar and bookmarking very easily. His test had the bookmarks feature available. He also was able to easily go to the event pages. The organization of the results also seemed intuitive to him. He also easily grasped the Critics' Pick notification. He did not reference the map very much, but granted, Stephanie is not picky about where she goes.

Notecard:



Takeaways:

The bookmarks feature is a great development. Also, we decided to lock the map on the left side of the screen and display the results on the right side. That way, when the user is scrolling through the results, they can always see the map and know where spatially any event is located. This feedback came from User 1's test. We decided to leave the searchbar relatively untouched because both users easily grasped the search fields and flow of action. We decided not to implement a search field by time because different events' times are displayed irregularly, and our users did not have trouble finding times once they selected events they were interested in. Both users were first interested in the event, and then in the time at which they take place. Both users found the landing page simple and easy to grasp, so we left that layout largely untouched (albeit sprucing it up a bit).

Second Set of Tests:

Third Test User: Kaixi as Greggory Granger and Stephanie Lovegood

She did not make the connection between the text "New York Times Critics' Pick" and the star image on the card. Otherwise, the test went smoothly: she was able to perform appropriate searches, navigate the map, and manage her bookmarks.

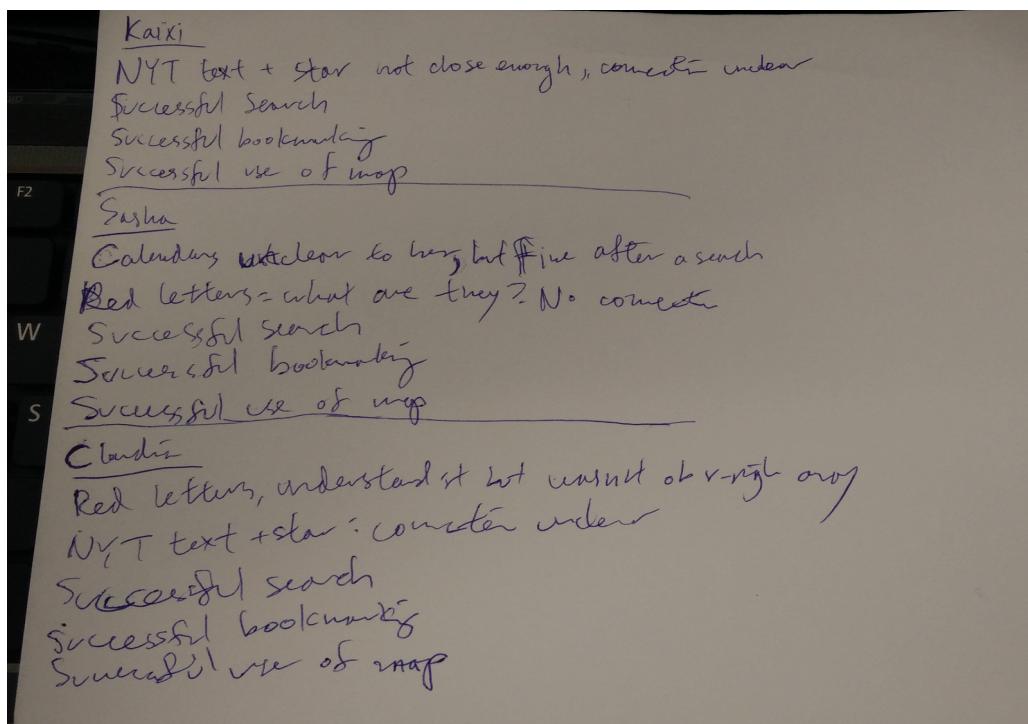
Fourth Test User: Sasha as Stephanie Lovegood

She did not make the connection between the red letters on top of the cards and the letters on the pins on the map. Initially she had difficulty with selecting dates and was not clear what each calendar widget represented in terms of the search but figured it out after performing a search. Otherwise, the test went smoothly.

Fifth Test User: Claudia as Greggory Granger

She did not make the connection between the text "New York Times Critics' Pick" and the star image on the card. She did not make the connection between the red letters on top of the cards and the letters on the pins on the map. Otherwise, the test went smoothly.

Notecard:



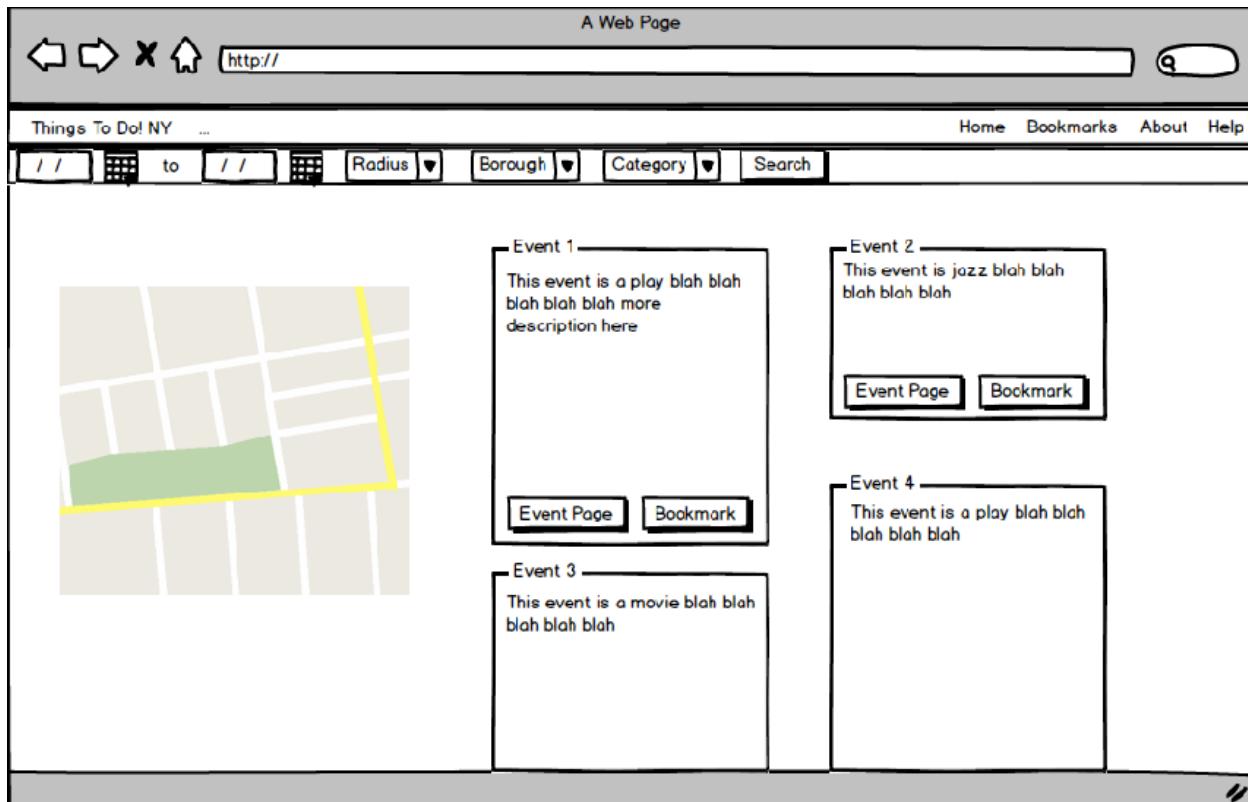
Takeaways:

We made small changes to the UI to improve clarity. We changed the red letters on the top of the cards to the same pins as those on the map so the connection between the two could be readily made (reinforcing the heuristic of consistency and standards). We added "start" and "end" labels to the calendar widgets to clarify what they represent in terms of the search. We styled the star and the text "New York Times Critics' Pick" to be visually closer to each other so their collective meaning could be readily understood.

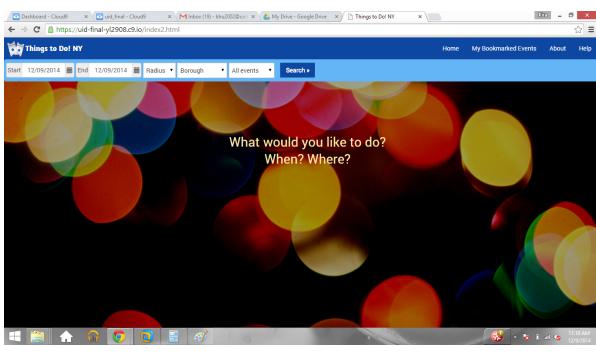
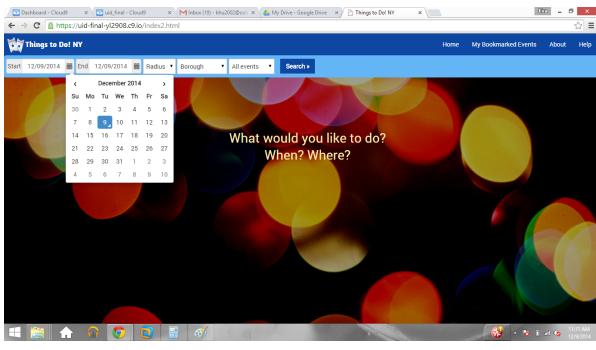
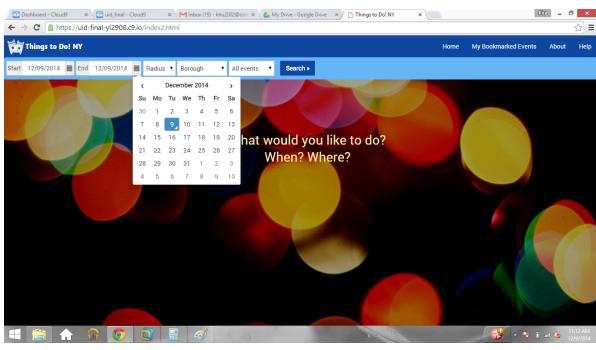
Final design:

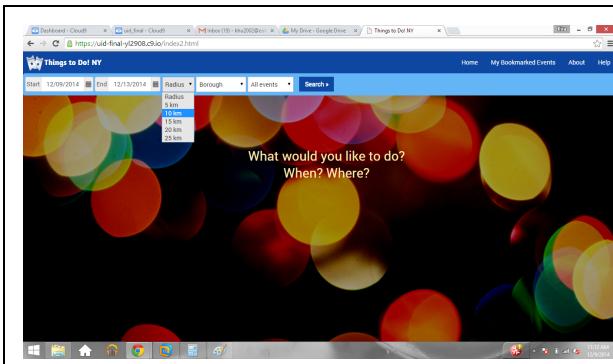
After testing, this is the final design template (done in Balsamiq) that we settled on.

We add options to search by borough and by category. We also lock the map on the left side of the page, and each result has a letter that corresponds to a label on the map. We also added "About" and "Help" links in the upper right corner to help users in case they get stuck. We also decided to go ahead with the bookmarks idea, and the list of bookmarks will be linked from a button also in the upper right corner. When appropriate, previous and next page buttons are visible on the search bar.

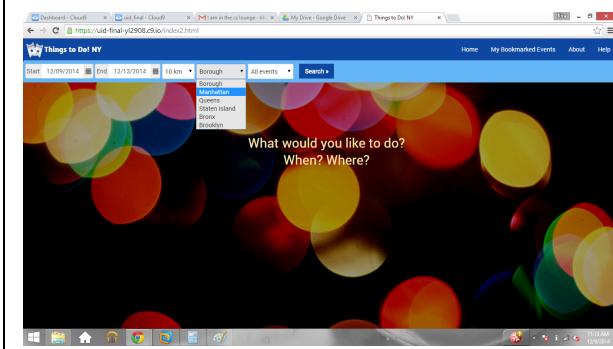


Final Storyboard:

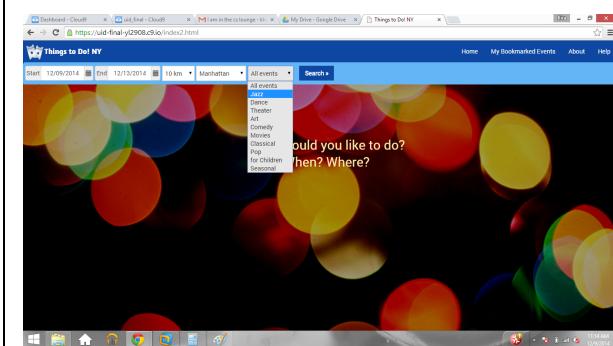
	<p>Landing Page</p>
	<p>User clicks “Go!” button</p>
	<p>Select the start date with the drop-down calendar. This field is not required to change, which means that if the user does not select a particular start date, the default date will return valid search results.</p>
	<p>Select the end date with the drop-down calendar. This field is not required to change.</p>



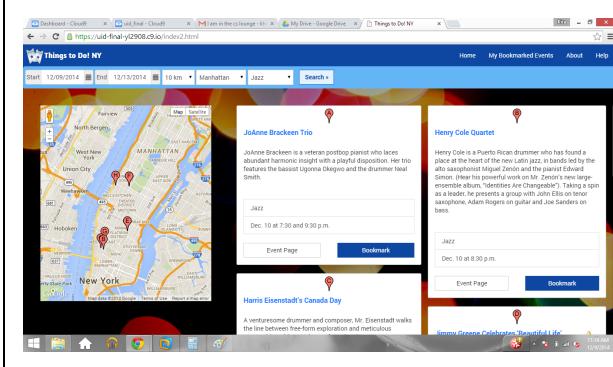
Select the radius from the drop-down. This field is not required.



Select the borough from the drop-down. This field is not required.



Select the category of events. This field is not required.



Click Search

Scroll down the page - the map stays locked on the left.

If more than 20 results are available, next page option appears in top right.

Click Next

Click Previous

Click on a letter in the map

The screenshot shows a map of Manhattan with various event locations marked by red pins. A tooltip for the 'Alvin Ailey American Dance Theater' event is displayed, containing text about their production of 'The Nutcracker'.

Click on event page - this could happen from the map, from the bookmarks window, or from any individual result.

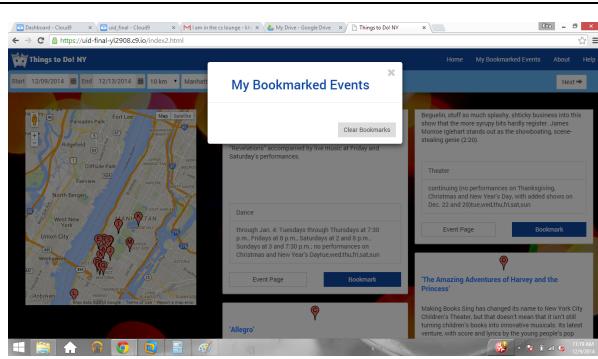
The screenshot shows a news article from The New York Times under the 'TOP FIVE' heading. It features a small map and a link to the 'Alvin Ailey American Dance Theater' event page.

Click “Bookmark” on the Alvin Ailey event.

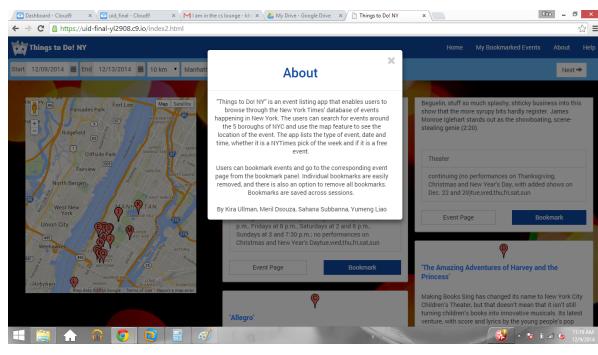
The screenshot shows the 'My Bookmarked Events' window on the Things to Do NY site. The 'Alvin Ailey American Dance Theater' event is listed with a 'Bookmark' button.

Click “remove bookmark” on the bottom entry.

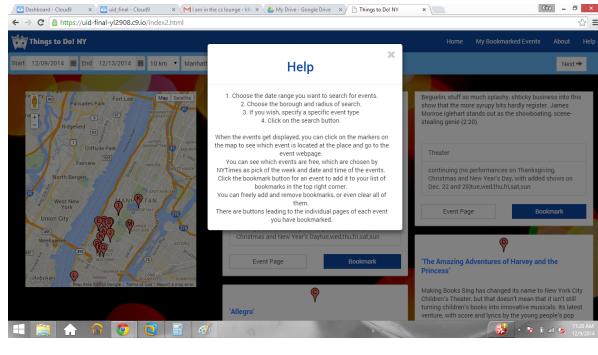
The screenshot shows the 'My Bookmarked Events' window again, but the 'Alvin Ailey American Dance Theater' entry is no longer present, indicating it has been removed.



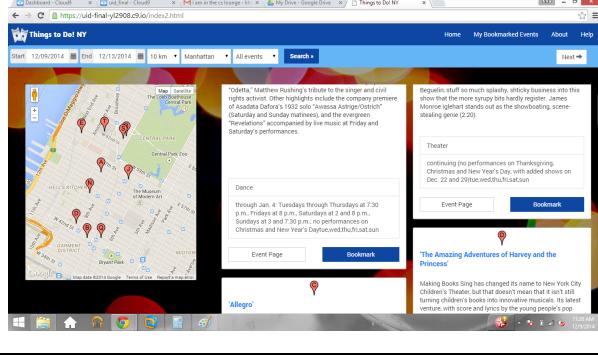
Click “Clear Bookmarks.” The bookmarks window will close, but if you check it again it will be empty.



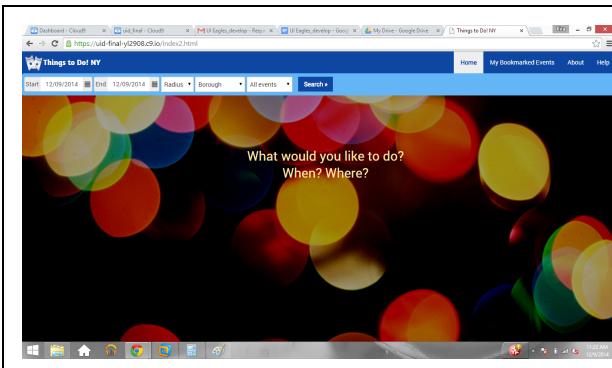
Click “About”



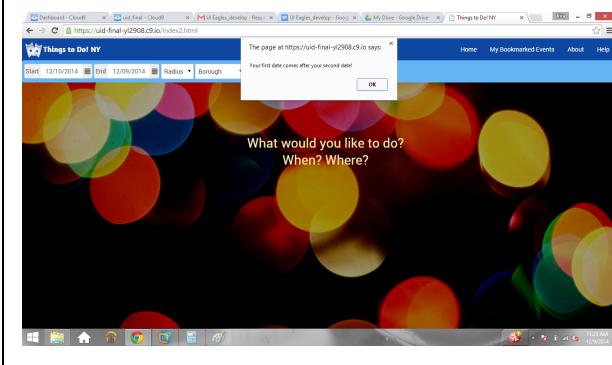
Click “Help”



Navigate within the map.



Click on the “Home” button



An alert pops up if the user selects a start date that is after the end date.

5. Software Engineering:

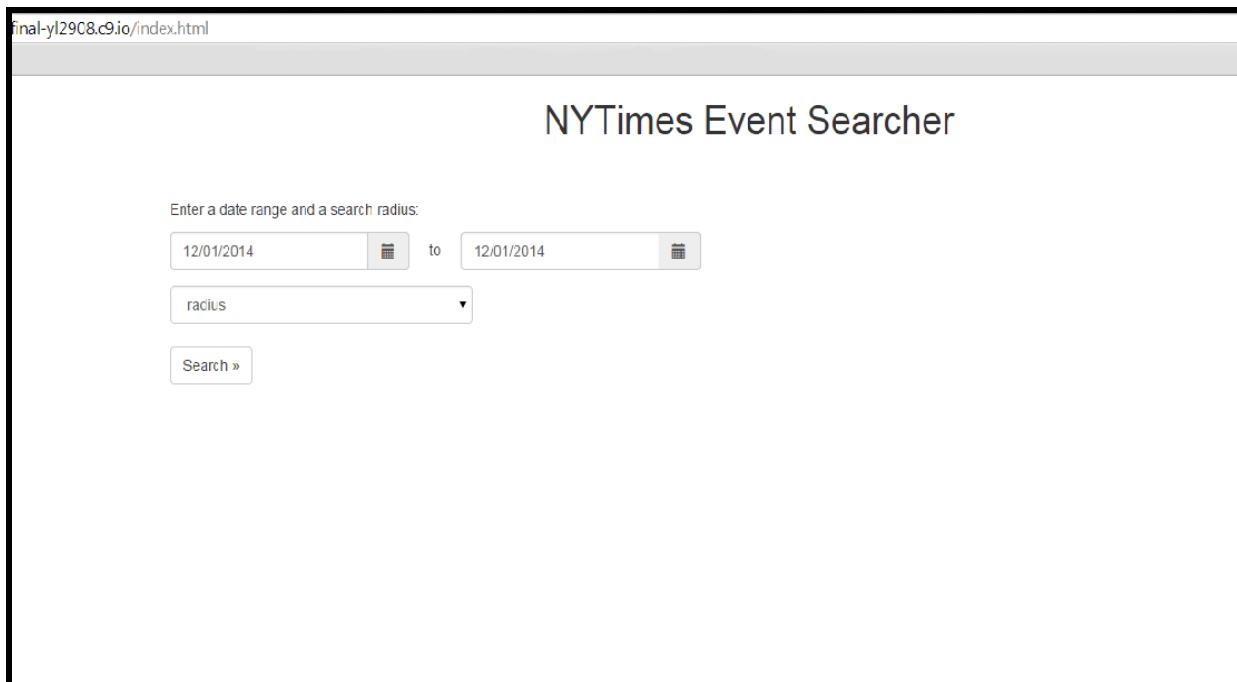
We used Bootstrap 3.2.0 to create the backbone of the UI, and various types of CSS (overriding style sheets, inline styling) to style the various aspects of the UI.

To allow the user to choose dates, we used the latest version of the calendar widget bootstrap-datetimepicker (<https://github.com/Eonasdan/bootstrap-datetimepicker>).

For backend, we used plain Javascript and Jquery 2.1.1. To get relevant data, we used the NYT Event Listing API to find events and the Google Maps API to display them.

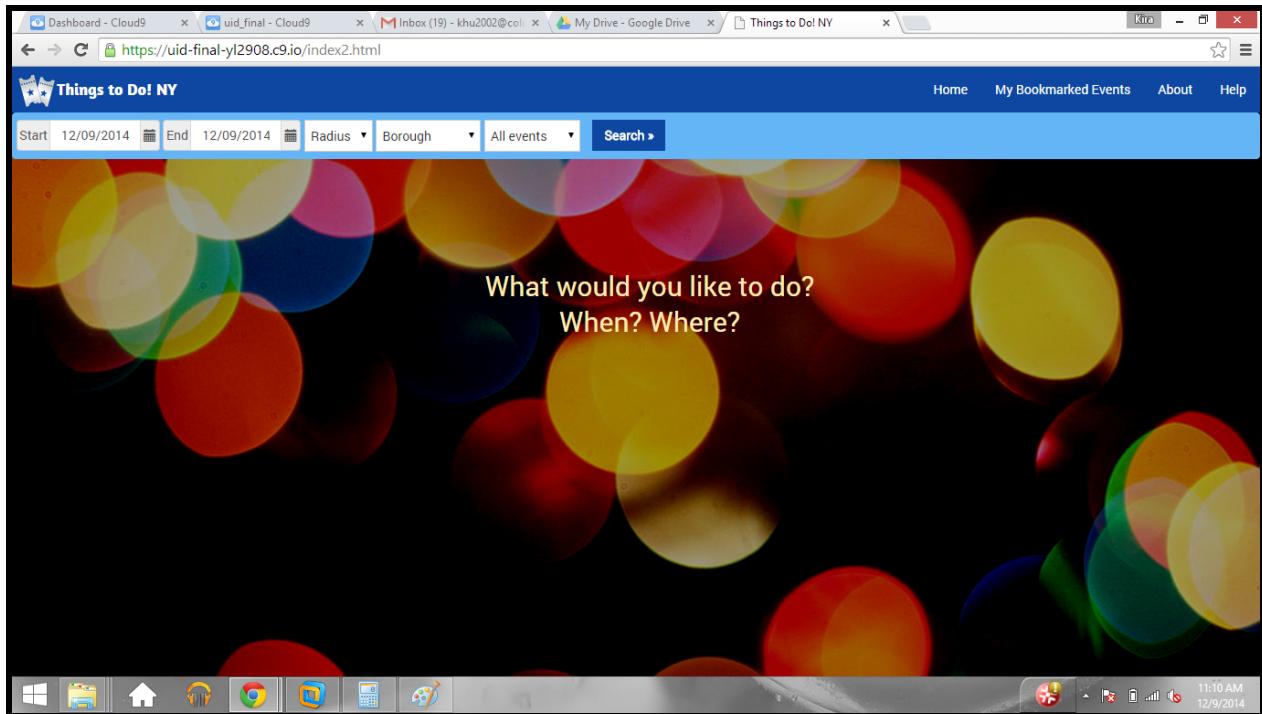
For the across-session bookmark bar, we used HTML5 Storage.

First version of web UI:



The screenshot shows a web browser window with the URL "final-yi29c8.c9.io/index.html" in the address bar. The main content area is titled "NYTimes Event Searcher". Below the title, there is a form with the placeholder text "Enter a date range and a search radius:". The form contains two date input fields, each with a calendar icon, separated by the word "to". Below these fields is a dropdown menu labeled "radius" with a downward arrow. At the bottom of the form is a button labeled "Search »". The entire screenshot is enclosed in a black rectangular border.

Final version of web UI:



Video Links

Youtube link: <https://www.youtube.com/watch?v=yEDHwn4CfMc>

Note: Youtube compresses the video to a very low quality, Google Drive link will retain high quality. Need to be signed in to Lionmail to view Drive link.

Google Drive link:

https://drive.google.com/a/columbia.edu/file/d/0B-0kgVKmT_G2V0xleDNaaW9YYmc/view?usp=sharing