
Admin Panel Customization in Book-a-Nook

Completion Report

Google Summer of Code 2016

Organization: Berkman Klein Center for Internet and Society

By: Shubham Patel (shubhpatel108@gmail.com)

Mentors:

Jessica Yurkofsky (jessica@metalab.harvard.edu)

Justin Clark (jclark@cyber.law.harvard.edu)

Project Proposal: <http://shubhpatel108.github.io/gsoc/proposal.pdf>

Code: <https://github.com/berkmancenter/bookanook/commits/dev?author=shubhpatel108>

Repository: <https://github.com/berkmancenter/bookanook>

Demo Video: <https://www.youtube.com/watch?v=s4ZgtZ7T7b8>

Terminology

1. Location = Library
2. Nook = Room
3. Superadmin = Universal Project manager, may or may not be related to any Library
4. Admins = Library Staff
5. Patrons = General users

About Book-a-Nook

Book-a-Nook is an open-source platform aiming to activate Library spaces. It provides a simple yet powerful platform for patrons to check available spaces and place a reservation request. The Admin panel allows admin to easily resolve the conflicts and approve/reject these requests. Another goal is to derive statistics for a better understanding of such demands and its variations.

Achieved Targets:

1. Travis and CodeClimate Integration
 2. Role management using Rolify
 - a. Creation of roles: Superadmins, Admins, Patrons
 - b. Association of Admins and Locations
 3. Resource access management using CanCanCan
 - a. Full access to Superadmins, including access to Users
 - b. Access to associated Locations and Nooks for Admins
 - c. Read access to Locations and Nooks for Patrons
 4. Authentication through Google+ using Omniauth
 5. Admin Calender using FullCalender.js
 - a. With three views - monthly, weekly and day
-

-
- b. One click reservation creation for week and day view- will open reservation form with prefilled date and time
 - c. Usage of colors to represent reservations with different status
 6. One-click conflict resolution of requests for admins
 7. One-click approval or rejection of reservation request
 8. Welcome dashboard in Admin panel to put focus on:
 - a. Reservations happening today
 - b. New reservation requests
 - c. Recently canceled reservations
 9. Expanded-nook view for Patrons
 - a. to allow comparison of Nook's availability through the entire week
 - b. to allow cross-comparison of all Nooks' availability at a particular time
 - c. with closed hours and reserved slots blocked as unavailable
 10. Google Maps Integration
 - a. Allow admin to mark the geolocation on Google Maps when creating and updating Locations
 - b. Use the geolocation to show distance of each Location from Patron
 - c. Show the Location's map on Nook's profile
 11. Flagging and adding remarks to reservations. E.g. no-show, lost keys.
 12. Enable Booking on half-hour basis (e.g. 8:30am to 12:3pm). Originally we allowed reservation requests on an hour basis (e.g. 8:00am to 1:00pm)
 - a. Integrate it with expanded-nook view for Patrons
 - b. Integrate it with new and edit reservation forms for Patrons
 - c. Integrate it with new and edit reservation forms in Admin panel
 13. Report dashboards
 - a. Nooks-stats dashboard - Reservations statistics related to Nooks
 - b. Location-stats dashboard - Reservations statistics related to Locations
 - c. Both the dashboard with following charts (implemented using Highcharts)
 - i. No. of reservations per month per Nook/Location
 - ii. No. of hours of reservations per Month per Nook/Location
 - iii. Aggregated no. of reservations at a particular hour of the day on all dates in selected range
 - iv. Aggregated no. of reservations at a particular hour of the day on a particular selected weekday.
 - v. Aggregated no. of reservations on all 7 days of the week
 - d. Downloadable charts (implemented using Highcharts)
 - e. Both the dashboard with filters to select Nooks/Locations, start-date and end-date.
 - f. Downloadable spreadsheet containing reservations

14. Email Notifications

- a. To Patron when Admin updates his/her reservation request's status. e.g. approved or rejected
- b. Reminder email to all users with reservations happening on a particular date
- c. Designed template with "Add to calendar" links when reservation is approved. Calendar types include - Google, iCal, Outlook and Yahoo.

15. Dashboard Notifications

- a. Send notifications to Patron when:
 - i. When Admin updates his/her reservation request
 - ii. When Admin approves/rejects/cancels his/her reservation request
 - iii. When Admin makes a reservation request on his/her behalf
- b. Notification archive to allow Patron to see old notifications

16. Restrict when Patron can make reservation request

- a. before 2 hours to the event [default]
- b. not before 2 months to the event [default]
- c. Make above values Nook-specific and modifiable by Admins

17. Introduce an upper cap on number of hours a Patron can request for in one day

18. Integration with AWS S3 to store the Nook's images

19. Progress Bar in Admin panel

20. Use Glyphicons instead of edit and delete buttons in Admin panel

Missed Targets:

1. Internationalisation through Multiple timezone support
 - Reason: Under-estimation of the task's complexity and required time
2. Google Calendar API integration
 - Reason: Prioritization of other tasks
3. Allow Patron to favourite Locations or Nooks
 - Reason: Prioritization of other tasks

Stats:

- Lines of Code written: 7238++, 1122--
 - No. of commits: 79
 - Pull requests (merged): 23
 - GitHub issues created: 81
 - GitHub issues solved: 98
-

Acknowledgement

Million thanks to **Jessica Yurkofsky**, my mentor, for her exemplary and extraordinary support throughout the program. It could have been very difficult to understand the project requirements without her constant support through quick response on emails, video chats, Google Hangouts, and GitHub issues! Thanks for providing quick mockups to instill a better idea of the product. Thanks for completing the feedback loop by testing the app thoroughly after every feature release. Thanks for the great support when I was not fully available for the development. Much gratitude for many other things I forgot to write here!

Million thanks to **Justin Clark**, my technical mentor, for providing zero-latency technical support by solving many implementation doubts. Thanks for trusting me and letting me work directly on the default branch of the repository!

Special thanks to **Krystelle Denis** for designing awesome UI mockups and **Paulina Haduong** for providing great ideas through video chats. Thanks to all people at Berkman Klein Center for the awesome support and for providing this great opportunity to be a part of this organization.

My gratitude to **Google Open Source Office** for organizing GSoC and for including me to be a part of the global open source community. My thanks to all organizer involved at every level!

Thanks to all who has helped in making this project a success!
