

Grizzly Hacks / Team HoneyPot

1. Features

Implemented Features

- Card on Events page.
- Featured Events expand
- Non-edu users can login
- Mobile view for Events page
- Judge tracker added
- Manage timeline/schedule feature added

Remaining Features

- User dashboard does not show non-edu users seeking a fix.
 - Blog may want to be renamed to "Event" website has events in the header but the logic in the codebase still refers to "blog"
 - Seek out with the approval of the officers a color palette that is appealing and readable for visually impaired users.
 - Judges tracker has not been synced to the firebase database.
-

2. Known Issues

- User dashboard does not display non-edu user info
 - CSS is all over the place from previous groups may want to redo and make a global CSS
 - Projects page needs to be fully implemented.
 - The Projects section is not working correctly.
 - Fix issues where users can not see errors when submitting projects due to CSS styling.
-

3. Project Flyer



Meet the Team

Steven Lopez

William Randazzo

Derron Pierre

Isaiah Jimenez

4. Screencast Link

<https://youtu.be/PQhXv2pKJ5I>

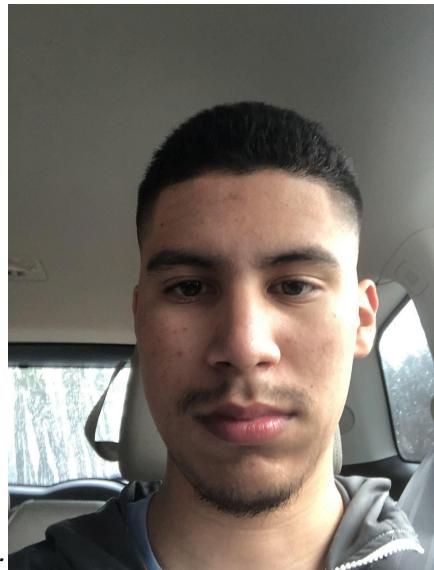
5. Team Members and Client

Team Members

- **Steven Lopez** - I am the project lead and also am a code architect.
- **William Randazzo** - I am the UI/UX design lead, and client liaison
- **Derron Pierre** - I am the documentation lead and the data modeler.
- **Isaiah Jimenez** - I am the lead tester and the other half of the code architects



WR:



IJ:



SL:



DP:

Client

- **Client Name:** Grizzly Hacks Officers
- **Brief Bio:** The Grizzly Hacks officers facilitate and organize any software related event on and off the GGC campus, primarily workshops and hackathons.
- ***Client photo:***



6. Project Abstract

This semester, my team and I worked on improving the Grizzly Hacks website. We met regularly with the Grizzly Hacks officers, Dr. Anca, and Dr. Gunay to get feedback and make updates. Most of our work focused on cleaning up the pages, making the UI more user-friendly, and adding new features—like a timeline page to help users keep track of important dates. It's been a great learning experience and a solid chance to apply what we've been learning in class to something real.

7. Testing

Testing Coverage, Methods, and Results

- **Unit Testing:** Jasmine + Karma (Angular CLI)

- **End-to-End Testing:** Cypress 12.x
 - **Environment:** Local server (<http://localhost:4200>), Firebase backend
-

Test Results Summary

Test Type	Tool	Executed	Passed	Failed	Result
Unit Tests	Jasmine + Karma	100%	100%	0	Full Pass
Acceptance Tests	Cypress	5	5	0	Full Pass
Integration Tests	Cypress	4	4	0	Full Pass
System Tests	Cypress	4	4	0	Full Pass

Unit Testing

- Ran with `ng test` (Karma + Jasmine).
 - All component/service logic verified and passed.
-

Cypress Testing

- Acceptance: Homepage, Footer Year, Navbar Events, Create Account Navigation, Title Verification.

- Integration: Navbar + Footer Display, Events Page Navigation, Title Metadata.
 - System: Overall page loads, UI checks, Navbar/Footer consistency.
-

Known Issues

- Minor JavaScript error (`featureScript.js`) during scroll detection.
- Suppressed in Cypress using:

```
javascript
CopyEdit
Cypress.on('uncaught:exception', (err, runnable) => { return
false; });
```

- Fix issue with (`verifyemail.component`)
 - No critical impact on functionality or user experience.
-

Recommendations

- Add programmatic login for testing authenticated routes (Projects, Admin Pages).
 - Expand form submission tests (Contact, Event Creation).
 - Integrate Cypress headless tests (`npx cypress run`) into CI/CD.
 - Review and harden front-end custom scripts to prevent DOM errors.
-

Status

- All Unit, Acceptance, Integration, and System Tests Passed.
 - Application stable and test-ready for future development.
-

8. Documentation

Installation Instructions

- The installation instructions can be found here:
 - <https://github.com/GGC-SD/GrizzlyHacks/blob/master/README.md#-project-installation-steps--how-to-run>

Help Page(s)

- Troubleshooting pages can be found here:
 - <https://github.com/GGC-SD/GrizzlyHacks/blob/master/README.md#troubleshooting-common-issues>

Code Documentation

- The code docs can be found here:
 - <https://github.com/GGC-SD/GrizzlyHacks/blob/master/README.md>
-

9. Usability Testing and Focus Group Survey

Overview

Usability testing for Grizzly Hacks focused on evaluating how easily users could navigate the site, and find event information.

Key Testing Areas

- **Navigation:** Testing whether users could easily find major pages (Home, Events, Timeline).
- **Mobile Responsiveness:** Ensuring a consistent experience across devices (desktop, tablet, mobile).
- **Login/Register Flow:** Checking the clarity and success rate of account creation and login processes.

Major Findings

- **Positive Feedback:**
 - Users found the event timeline intuitive and easy to follow.
 - The site's overall design was considered clean and student-friendly.
- **Areas for Improvement:**
 - User account creation was limited to 5 new sign ups, this is a Firebase issue due to the current plan we are using.
 - Some minor contrast issues on desktop and mobile due to bad contrast.
 - Project submissions need to be fixed.

Changes Implemented

- Adjusted CSS to fix spacing issues for smaller screen sizes.
- Enhanced the homepage to feature upcoming event banners more clearly, based on feedback.

Overall Summary

Usability testing showed that Grizzly Hacks were generally easy to use and well-received. Most improvements needed were minor tweaks for mobile optimization and faster content loading. Feedback from testers directly influenced the final polish of the project, especially on responsiveness and visual clarity.

10. Documentation Locations

Where to find:

- **Installation Guide:** [Click here](#)
 - **Developer Documentation:** [Click here](#)
 - **User Documentation:** [Click here](#)
-

11. License and Intellectual Property Terms

Software License

Copyright 2025 Team Honeypot

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Intellectual Property Terms

[Click here to visit](#)

Signatures

Name	Percent	Signature
William Randazzo	15%	WR
Steven Lopez	15%	SL
Derron Pierre	10%	DP
Isaiah Jimenez	5%	IJ
Client/Officers	55%	CO

12. Sprint Velocity Report

Sprint 1

- Weekly velocity:
- William: the first iteration could have gone better the velocity chart is lacking because mainly I forgot to update it as I continued our work
- Notes on performance:
- William : I feel as if I performed adequate with what was expected of us, however I could have utilized JIRA more

-Derron

- Weekly velocity
- The first iteration in my opinion was a little tough since this was my first time working like this, most of the time I would work on something and not update my teammates or JIRA.

- Notes on performance:
- This was the setting stone for the other two iterations I learned alot from this behind the feedback of DR Anca and my teammates.
- Steven
 - Weekly velocity: The velocity chart wasn't as strong as it could be but this was mainly due to the fact I didn't correctly estimate story points and we would forget to track our tickets.
 - Notes on performance: I had much to improve on this Sprint as the Project Lead, such as correctly estimating story points and creating and adding all the tickets to the Sprint that were needed instead of adding them after creating the Sprint.

Isaiah Jimenez:

- Weekly velocity: That was my first time using Jira and it took me a while to understand its features, but once my PM assigned us tickets and tasks, I was able to understand what I needed to do and use certain features to update the progress. Although like mentioned by my team members the first sprint was a learning experience for all of us.
- Performance Notes: I wished I could have included screenshots of my progress to show the before and after, in my tickets. That would have helped show my activity better, on my end towards my team members.

Sprint 2

- Weekly velocity:
- William: This iteration went a lot better. I had committed a lot more than usual and I had been actively updating tickets throughout the whole project.
- Notes on performance:
- William: I think this sprint was more efficient thanks to me utilizing JIRA a lot more and properly allocating story points.
- Derron
 - Weekly velocity
 - This iteration was better than the first, as we learned from our mistakes and started updating the JIRA board more.
 - Notes on performance
 - My performance took a step in the right direction. As I was able to work on tickets and close some of the tickets that had been backlogged from the first iteration.
- Steven
 - Weekly velocity: There was significant improvement in this sprint due to the fact we learned from the mistakes made in our first sprint.
 - Notes on performance: I think I did much better on this sprint as the project lead, I am much more content with this performance compared to the 1st sprint, but of course there is still much to learn.

Isaiah Jimenez:

- Weekly velocity: After the first iteration, and the errors we made, we organized this sprint much better. Our PM helped us as well with the organization and implementation. This

helped us tremendously because we knew exactly how to show more activity and progress in our sprint. And it showed in the end as our progress spiked.

- Performance Notes: I felt much more confident when using JIRA, I knew as soon as I moved from one stage to another to update the progress of that task. One thing I wished I knew during this sprint that would have helped me a lot, was how to see all the tickets that were assigned to one person. I had to scroll down throughout all the tasks and personally click on them and see if it was assigned, this was very tedious.

Sprint 3

- Weekly velocity:
- William : This was the 2nd best iteration in my opinion as we wrapped up it helped me identify bugs and errors we got and leave them in the comments for the appropriate file for the next team.
- Notes on performance:
- William : This was a good clean up sprint where I could focus on getting some backlog cleared as well as wrap up the ongoing work I had to do.
- Derron
 - Weekly velocity
 - This iteration, in my opinion, was our best as a team. We wrapped up a lot of tickets that had been opened and were also allowed to fix any bugs we encountered.
- Notes on performance
- My overall performance on this iteration was pretty good. Being able to wrap up some of my bigger tickets and also fix all the errors that were bothersome helped me grow as a programmer.
- Steven
 - Weekly velocity: Things simmered down a bit in the final sprint as we were wrapping up existing tickets from Sprint 2 which is why I think the velocity chart here reflects that.
 - Notes on performance: I became much more comfortable with Jira and managing Sprints/teams by this iteration which I think is a great stepping stone and will benefit me greatly in my professional career.

Isaiah Jimenez:

- Weekly velocity: Not as much to do in this sprint since most of it was done in the previous sprints, however we worked on several features mainly cleaning up and testing (on my end). Also adding and fixing components to make the project more polished.
 - Performance Notes: Compared to my first time using JIRA, I personally witnessed my growth and understanding. I now feel like if I have to use JIRA again, I know exactly what to do.
-

Lessons Learned

- What went well:
 - William: I feel as if our teamwork got us through this project because of how much we collaborated
 - Derron: As a team, we communicated well throughout the semester. Our Discord chats helped keep all of us in tune and made this project not as hard as I thought it would be.
 - Steven: As the semester progressed I believe our team communication grew and got much stronger. We became very chalant with each other which led to smoother iterations and discussions.
 - Isaiah Jimenez: I felt like the cohesion and completion of our work as a team went really well. We knew what we needed to do, and we did it. My teammates also helped each other out when needed.
- What went wrong:
 - William: I feel as if we could have met a bit more than we did
 - Derron: I felt like some tasks took longer to complete than they should have.
 - Steven: I think accountability could have been done better, as the project lead I should have instituted some sort of daily check-in to better track progress of tasks.
 - Isaiah: I felt like I needed to understand better how certain components and features worked, in order to fix/improve them. I found myself struggling often to adjust some basic features. And as a code architect, you need that.
- How we improved:
 - William: We got on each other and allowed each of us to improve our standards and practice while working on this project
 - Derron: I felt like we grew so close during the semester that we were able to call out to each other if one of us was slacking, get them back on track.
 - Steven: After each iteration we seemed to get better at knocking out tasks and we improved this by communicating more.
 - Isaiah: As the weeks went by we grew into our roles better, and along with that our communication. We helped each other when we were struggling with tasks.

- What we would do differently next time:
 - William : maybe just get together more so we could focus on bigger tickets and maybe even solve one big issue together.
 - Derron: I think next time we should hold more voice chats. I think that would help us a lot, so we can discuss what we were doing and what needs to be done. And also, if anyone needs help.
 - Steven: As project lead, I would have an established workflow written out to guide teammates, I would better estimate story points, and I would have daily quick meetings to check in on the progress of everyone's issues.
 - Isaiah: I would have needed to put more of an effort to understand component structure, to better help in my role as code architecture. And with that I could have also improved my progress with testing.