

Project Closure and Recap Form

This form should be completed after reaching the final milestone and upon completing your project at the end of the semester. A single form must be submitted by each team.

First, please fill the following entries with information about your project:

- **Group #:** 2
- **Group Members:** Alex Hoff, Ethan Day, Hannah Klein, Max Benish, Nico Bacic
- **Site URL:** <https://hdl1-training-statuses.web.app/>

Next, engage in self-reflection by addressing the following questions:

1. What worked well, and why?

We did a good job communicating. Each member in our group was very active and willing to participate and communicate with one another. We had a group chat that we would very consistently use and everyone was willing to jump on a call or meet in person if we needed to. We all believed in our project and wanted to do our part to make sure that we were producing the best version of our site.

2. What didn't work well, and why not?

Something that we noticed when working through the semester was it was hard for an equal amount of work to be done by each member. We all have different schedules and different coding strengths, so it was hard to make sure that we all did equal work. Certain members excelled at coding while other members of the group excelled in project management. We tried our best to balance the amount of work, but at the end of the project it wasn't completely even.

3. If you were to do the same project again, what would you do differently?

Something we would do differently would be to have a better understanding of firebase and the layout of our collections/documents before creating them. We have a very complex database with many documents and sub collections. One worry that we have is the amount of reads/writes our database goes through when running our site. If we were to complete this project again we would try to factor firebase rules about maximum writes/read into our design so that we don't get charged if our site begins to exceed these maximums.

4. What are your takeaways after completing the project?

After completing this project we have learned a lot about different project management styles and the pros and cons of each system. When participating in sprints we learned about the fast paced turnaround and the ability to meet deadlines. From each sprint we took away the importance of working on the project a little everyday to meet our goals and show our client our progress. From the overall milestones we felt it helpful that we got to spread the project out over the entire semester, which emphasized the importance of not procrastinating. Overall the opportunity to work in a small group provided valuable learning experiences about working with others and communicating.

5. *Describe a scenario that you thought would be hard to go through, but it turned out to be easier than originally thought. Describe another for the other way around.*

We scheduled a large part of sprint 3 to rework the main page's columnar layout into a single unified table that would handle screen resizing better. We expected this process to take most of the sprint, however we were able to complete most of the basic design in two days.

On the other hand, one of our biggest ideas starting out was to be able to drag and drop employee tiles between each of the training levels on the main page. As we got closer to implementation, we realized the complexity of that problem, as well as its overall unimportance - employees move in a linear path from one training level to the next, with no real need for manual input when they fulfill the training requirements.

6. *Have you showed your project to people outside your class? What was the impression?*

Most of us have shown the project to our roommates, who have unanimously been impressed. Compared to most classes, being able to see actionable, tangible results like the completion of this project makes the class seem all the more useful in the long term.

7. *What other features would you like to add if you have more time?*









Finding a way to implement the drag-and-drop functionality would definitely be on our list for a theoretical sprint 6. Additionally, something we may include anyways would be improved code legibility, comments, or a general walkthrough with the client of the code and how it works. One of our concerns post-project is whether or not the help desk managers will be able to implement fixes or potential updates to the system, however this is outside the scope of the project.

8. *Anything else you want to say?*

We enjoyed the challenge and complexity of such a sophisticated database design, especially considering it was our first time designing a nonrelational database. Once again, being able to see the project come together from both a design and functional perspective over the last few weeks has made it all worth the effort.

Lastly, capture multiple screenshots from your personal Firebase account to illustrate the following points. Attach your screenshots after the corresponding items below.

- 1. Show your existing users.

| Identifier | Providers | Created ↓ | Signed In | User UID |
|-------------------------|---|--------------|--------------|------------------------------|
| amhoff2@wisc.edu |  | Nov 10, 2024 | Dec 5, 2024 | n7MLfUxkdcWuhX5EcAGAKd... |
| erday2+test@wisc.edu |  | Nov 9, 2024 | Nov 9, 2024 | Wd4Rq9MjRnRV4Nw6OGglA1... |
| ahoff2910+testuser@g... |  | Oct 15, 2024 | Oct 15, 2024 | Nu9XrDIM7ieROsQrT9KM4E9... |
| ahoff2910+testadmin@... |  | Oct 15, 2024 | Nov 12, 2024 | laVxHKYrHwWpuHKfzMNVM9... |
| bacic@wisc.edu |  | Oct 11, 2024 | Nov 8, 2024 | i7fk9UiHUhWnAByliEcMehARi... |
| mrbenish@wisc.edu |  | Oct 11, 2024 | Dec 3, 2024 | s5CWfUWmn9O83yJF0Nmvq... |
| erday2@wisc.edu |  | Oct 11, 2024 | Dec 5, 2024 | 1WzOcwK9AsMpxqfrRwlJHuF... |
| hbklein@wisc.edu |  | Oct 11, 2024 | Oct 13, 2024 | auf2QczZ6tgch5yaqXaXNCLX... |

- 2. Show a comprehensive list of all your collections.

| | |
|---|--|
| <div><div>+ Start collection</div><div>agents ></div><div>training</div><div>users</div></div> | |
|---|--|

3. Show a sample document from each of your collections. For example, if you have 3 collections, provide 3 corresponding screenshots.

Agents active:

| | |
|---|--|
| <div>agents</div> <div>+ Add document</div> <div>active ></div> <div>departing</div> <div>inactive</div> | <div>active</div> <div>+ Start collection</div> <div>data</div> <div>+ Add field</div> <div>AANA</div> <div>agent: "AANA"</div> <div>flag: 0</div> <div>graduation: "Fall 2024"</div> <div>hireDate: "2024-08-15"</div> <div>lastReviewed: "2024-09-08"</div> <div>slp: 0</div> <div>status: "active"</div> <div>trainingLevel: 2</div> <div>trainingStatus: 0</div> |
|---|--|

Agents active subcollection:

| | | |
|---|--|--|
| <div>active</div> <div>+ Start collection</div> <div>data ></div> <div>+ Add field</div> <div>AANA</div> <div>agent: "AANA"</div> <div>flag: 0</div> <div>graduation: "Fall 2024"</div> <div>hireDate: "2024-08-15"</div> <div>lastReviewed: "2024-09-08"</div> <div>slp: 0</div> <div>status: "active"</div> <div>trainingLevel: 2</div> <div>trainingStatus: 0</div> | <div>data</div> <div>+ Add document</div> <div>AANA ></div> <div>AARK</div> <div>ACRI</div> <div>AFZA</div> <div>AGUR</div> <div>AHOD</div> <div>AKAU</div> <div>AKKI</div> <div>AMBE</div> <div>AMCU</div> <div>AMER</div> <div>AMHO</div> <div>ANIK</div> <div>APOR</div> | <div>AANA</div> <div>+ Start collection</div> <div>+ Add field</div> <div>agent: "AANA"</div> <div>flag: 0</div> <div>graduation: "Fall 2024"</div> <div>hireDate: "2024-08-15"</div> <div>lastReviewed: "2024-09-08"</div> <div>slp: 0</div> <div>status: "active"</div> <div>training</div> <div>Advanced Phones: {name: "Advanced Phones", ...}</div> <div>Pick 1: {name: "Pick 1", order: 1, ...} (map) +</div> <div>trainingLevel: 2</div> <div>trainingStatus: 0</div> |
|---|--|--|

Training

🏠 > training > training

More in Google Cloud

(default)

training

training

+ Start collection

agents

training

users

+ Add document

training

training

+ Start collection

+ Add field

Advanced Phones

name: "Advanced Phones"

order: 2

steps: [(type: 'checkbox', name: ...)]

Chat_Email

name: "Chat/Email"

order: 3

steps: [(type: 'checkbox', name: ...)]

Departing

name: "Departing"

order: 6

steps: [(name: "NOVA", type: 'che...)]

HDQA

name: "HDQA"

order: 5

steps: [(type: 'checkbox', name: ...)]

Onsite

name: "Onsite"

order: 4

steps: [(name: "Contacted Agent", ...)]

Pick 1

name: "Pick 1"

Database location: nam5

Users

🏠 > users > 1WzOcwK9AsM.

More in Google Cloud

(default)

users

1WzOcwK9AsMpxqfrRw1JHuFluUK2

+ Start collection

agents

training

users

+ Add document

1WzOcwK9AsMpxqfrRw1JHuFluUK2

3hLQpgf0NQSIegcCLVBqzTkYxsp1

5jS1ygF5UzVve1b5b1txsqoS2Ar2

IaVxHKYrHwWpuHKfzMNvrn9b4H32

Nu9XrDIN71eR0sQrT9KM4E9ANmr1

Wd4Rq9MjRnRV4Nw6OGgIA11koWH3

auf2QczZ6tgch5yaqXaXNCLXWWG2

i7fk9UiHUhWnAByl1EcMehArr1R2

n7MLfUxkdcWuhX5EcAGAKdRzoPV2

qk1b7UApQ3SYw24fQ12tp6c9BwF3

s5CWfUWmn9083yJF0NmvmvqOp10u1

+ Start collection

+ Add field

email: "erday2@wisc.edu"

name: "Ethan Day"

role: "admin"

4. Show your security rules configuration.

```
1 rules_version = '2';
2
3 service cloud.firestore {
4   match /databases/{database}/documents {
5     // Allow users to read their own user document
6     match /users/{userId} {
7       allow read: if request.auth.uid == userId;
8
9       // Allow users to update their own name and ensure the role remains the same
10      allow update: if request.auth.uid == userId
11                    && (request.resource.data.role == resource.data.role || !('role' in request.resource.data));
12
13      // Allow admins to write user documents
14      allow read, write: if request.auth != null
15                        && get(/databases/{database}/documents/users/{request.auth.uid}).data.role == 'admin';
16    }
17
18    // Allow signed-in users full access to the "agents" collection
19    match /agents/{agentId} {
20      allow read, write: if request.auth != null;
21
22      // Allow signed-in users to edit the "data" subcollection
23      match /data/{documentId} {
24        allow read, write: if request.auth != null;
25      }
26    }
27
28    // Allow admin users full access to the "training" collection
29    match /training/{trainingId} {
30      allow read, write: if request.auth != null
31                        && get(/databases/{database}/documents/users/{request.auth.uid}).data.role == 'admin';
32    }
33
34    // Block access to all other documents
35    match /{document=**} {
36      allow read, write: if false;
37    }
38  }
39 }
```