Final Write Up

1. Project Goal - A clear statement of the project goal
   1. Our goal for this project is to create a PTO system for the customer so that Employees can access their paid time off and specific days off such as VD, PD, and SD. As well as a view for the supervisor to see the days off they must approve or deny. They can also constantly update the record of the employees for those days off. We want to have a user-friendly system in place for the employees.
2. Stakeholders - All project stakeholders and their roles
   1. Professor David Lash : Supervisor
   2. Aldi IT Department : Customer
3. Deliverables - A deliverables definition – when your project is complete what specifically will you deliver? (Be clear, complete, and concise)?
   1. Our Deliverables Definition or our Definition of Done, is that it is good to submit when our supervisor approves of the work done on the backlog item.

Sprint 1 Review

Attendance: Professor Lash, Diego Rodriguez, Brian De La Torre, Issai Nunez, Anthony Salerno

1. Sprint 1 (October 28th – November 11th)
   1. Initial Sprint Backlog
      1. Final Front End Design
      2. Front End UI
      3. ER Diagram/Table Diagram
      4. Implementing Data from Tables to PHP Server
      5. Backend Development
   2. Items Completed
      1. Final Front End Design
      2. ER Diagram/Table Diagram
      3. Backend Development
   3. User Stories
      1. Final Front End Design
         1. Acceptance Criteria: Completing the design work for the log in page, the homepage, the PTO page, and the statistics page. Also completing designs for the lower-level employee view and the HR view.
      2. Front End UI
         1. User Story: As a user, when I enter the app, I want to see the login page which takes me to the homepage.
         2. Acceptance Criteria: Given that I am an employee. When I enter the application. I can log in with my employee id and password which will lead me to the homepage.

On the login page, I can also change my password in case I forget the original one.

When I click log in, it should take me to the homepage where I can see the date, a welcome user header, a button that will lead me to the times and absences page, and my notifications to the right.

Technical Requirements:  
On the login page a user can either put their information in, that is from the data, to log in or if they forget their password, they can click on 'Forget Password' and change their password.

Once they log in, they should see the homepage which should welcome them in the header. They should also see a button to click on which will lead them to their times and absences page, a notifications menu to the right side of the homepage, and their PTO statistics.

* + 1. ER Diagram and/or Table Diagram
       1. Acceptance Criteria: As the supervisor for this project, I want to see a diagram that is displaying at least 5 or more tables with each table containing a key.
    2. Implementing Data from Aldi into the PHP Server
       1. User Story: As the supervisor, with the data I provided to your team, I want to see the data shown in table form. I want to make sure you are putting the data in the correct labels.
       2. Acceptance Criteria: Outputting columns and rows of data using SQL within the PHP server with the Diagram created and the data from Aldi provided
    3. Smoothing Out Back End
       1. Acceptance Criteria: Getting the log in page to work on the backend as well as the register page for any new users.
  1. The Demo
     1. We will be showcasing our frontend work and the work of the backend. We will also be doing a demo of outputting the data.
  2. The Learnings
     1. The first thing we learned as a team was that we needed to give each other a more even amount of work to do for the application. It feels like we each need to have our own hands on this project so for the next sprint we need to spread out the coding more evenly. Like giving each of us a page to do.
     2. I think another thing we realized during the sprint was that we needed better communication with what the frontend and the backend was doing.
  3. Sprint Burn-Down
     1. So, everything for this sprint started out relatively smooth. Brian and Anthony already had a head start on coding. However, we were still iffy on a final design for the front end. But as Brian showed me his progress, we came to a final decision on the look of it. Things did become somewhat stagnant for Issai because there was an issue of trying to figure out what he should do for this sprint because he couldn’t be just backup. Thankfully we came to decide on him working on the data for our tables. But because of how I needed to get the diagram ready, he had a long wait which was a progress killer for a while, but we eventually pulled through. My diagram did eventually have to be reworked twice to make it simpler. Thankfully however, in the case that my ER Diagram will not be accepted Anthony had a table diagram ready to showcase. Once that was done, progress was going study even after a stagnant production wave.

Graphical user interface, application

Description automatically generated

* 1. Sprint Retrospective
  2. What Worked in Sprint 1
     1. We think what went well was the database coding on the backend with the authentication code from Anthony. As well as progress on the frontend with Brian going above and beyond what was planned for this sprint.
     2. We think the diagrams went well even though the ER diagram was not really used it helped us understand Anthony’s table diagram a lot better.
     3. We also think having meetings with each other helped us understand everyone’s work on the progress better as well.
  3. What needs work
     1. We think communication outside of our meetings need work
     2. The work needs to be better spread out among one another that way no one person is stuck on doing one entire part of the project
     3. When we hit a roadblock, we took too long to get around it. So, we need everyone on the team to become educated about what someone is working on to help a lot better
  4. What Will We Do?
     1. We will have better communication with each other outside of class and meetings
     2. Better use of GitHub because we are not too comfortable with it yet
     3. Get better acquainted with things like REACT and JSON to help and be able to work a lot more on those things. This coming from Diego and Issai

Sprint 2 Review

Attendance: Professor Lash, Diego Rodriguez, Brian De La Torre, Issai Nunez, Anthony Salerno

1. Sprint 1 (November 18th – November 30th)
   1. Initial Sprint Backlog
      1. Log In/Register Merge
      2. Requesting PTO Page
      3. Donut Charts
      4. Homepage
      5. Context Diagram
      6. Use Case Diagram
      7. State Diagram
   2. Items Completed
      1. Log In/Register Merge
      2. Donut Charts
      3. Homepage
   3. User Stories
      1. Log In/Register Merge
         1. User Story: As a user, I should be able to log in and register in with my employee ID and password, so that I can go into the PTO application.
         2. Acceptance Criteria: Given that I am an employee or a new user. When I enter the application, I should be able to see a log in page where I can enter my current employee credentials to log in and see the homepage. But, if I am a new employee, I should see a button on the login page that allows me to create new credentials for the application.
         3. Technical Requirements: On the log in page a returning user can put in the email they used when they created the employee account and password. They should see a page that shows two areas where they type in their email and password. If they type in something wrong, like the password, it will warn them that they are the wrong credentials. There is also a button underneath this that will ask new employees to create an account. Here they will be led to a page that asks for their email address and what the password will be for this new account.
      2. Donut Charts
         1. User Story: As a user, I want to be able to see my PTO data, how much I have left, and how much I originally had, in the form of Donut Charts.
         2. Acceptance Criteria: Given that I am an employee. When I enter the page which stores my PTO stats. I should see at most, 3 donut charts containing data for my Personal PTO, Sick PTO, and Vacation PTO. I should see the amount I have left and how much Time off I originally had.
         3. Technical Requirements: On the stats page, the user should see three donut charts displaying their PTO data. But for now, until we can get it working with our PHP server, let's hardcode in data that we can change to show that it can work with our application
      3. Homepage
         1. User Story: As a user, when I enter the app, I want to see the login page which takes me to the homepage.
         2. Acceptance Criteria: Given that I am an employee. When I enter the application. I can log in with my employee id and password which will lead me to the homepage. When I click log in, it should take me to the homepage where I can see the date, a welcome user header, a button that will lead me to the times and absences page, and my notifications to the right.
         3. Technical Requirements: Once they log in, they should see the homepage which should welcome them in the header. They should also see a button to click on which will lead them to their times and absences page, a notifications menu to the right side of the homepage, and their PTO statistics.
      4. Context Diagram
         1. Acceptance Criteria: As the supervisor of this project. I want to see a context diagram that is showing how the application interacts with the database, the employees, and the admins.
      5. Use Case Diagram:
         1. Acceptance Criteria: As the supervisor of the project, I should see a use case diagram that shows how employees and HR interact with one another on the application. How are the HR employee handling user requests
      6. State Diagram:
         1. Given that I am the supervisor, I want to see a state diagram showcasing the many events that the application will go through and the change in state PTO will make through the request process.
   4. The Demo
      1. From low priority to high priority order, we showed our diagrams which we then received feedback for. We then proceeded to show Professor Lash our login page working in by logging in and creating a new account. Brian would then show his work on the home page discussing what each button did and what the notifications displayed. Issai would showcase his work on the donut charts, which while they didn’t connect to our data server, it was a good showcase even with the hardcoded data and it showed exactly what we want to have on the final product.
   5. The Learnings
      1. From this sprint meeting we learned that when it came to the diagrams that it was important for us to go to Lash for help when something didn’t make sense or in case, we were doing the diagram wrong. Otherwise, we could turn in work that did not make sense or made things more complicated than it should.
      2. For the next and last sprint, we learned it was time to try and get the most important functionalities done. Otherwise, we will continue to have issues and have no time for them.
   6. Sprint Burn-Down
      1. Because of how a lot more focused this sprint became thanks to the feedback from the first sprint. We managed to put a lot more work into our frontend and backend functionality. Anthony practically worked on the daily when it came to the backend of our project and same went for Brian with our frontend. Both came together in the end, and we managed to get our log in and register pages working on the frontend and in our backend. Our requesting PTO page went through some changes in design during this sprint but overall, the principles such as the user story and requirements were the same. Brian did come to us with some changes he wanted to make, and they were good additions thankfully. Anthony was able to implement his backend to it and we managed to get a working PTO requesting system.

We also managed to get a final basis for our homepage. Making it simpler with some adjustments. Issai was on donut charts which we wanted to implement onto the stats page. He did manage to get those working kind of. But we ran into problem with our GitHub that made accessing the application impossible for the weekend. So, Issai had to wait while we tried to sort this out. But once we did, he got back to work as usual. In this sprint we also realized that we needed to get diagrams done since we only had the two, activity which wasn’t even needed anyway, and the table diagram.

So, everyone but Issai, since he already worked on a diagram, was assigned a diagram. Diego was assigned the context diagram, Brian the state diagrams for an employee view and an HR view, and finally Anthony had to work on a use case diagram. We did present those to Lash on the day of our meeting, but it looks like we need some more work on those. The context diagram was a little too complicated, state diagram was missing some things, and the use case diagram was a little incomplete. So, these must get carried over into our last sprint.

* 1. Graphical user interface, chart, application

     Description automatically generated
  2. Sprint Retrospective
  3. What Worked in Sprint II
     1. Backend and Frontend people had a lot more communication this time around. Brian and Anthony conversed a lot more and with that Anthony was able to better integrate his backend with Brian’s frontend code and design.
     2. Having us learn more about certain things we didn’t know anything about helped us with each other’s duties. Something like React was an issue so budding heads really helped.
     3. A lot more of helping one another with issues and clarification really helped us on this.
  4. What needs work
     1. Going to Professor Lash for help. When it came to the diagrams it’s clear that they were either too confusing, missing things, or too complicated. We need to go to him for any help on these so we can get a good diagram out there to better understand the application
     2. A better workload, this sprint even though the tasks were big. There seemed to be not too many of them, even with the diagrams.
     3. We need to continue talking to one another about the project even during breaks like weekends. Even though we do work on the project during those days. We seem to lose communication somehow
  5. What Will We Do?
     1. Meeting with Lash individually if we need to for help to get tasks completed and done right,
     2. Have a bigger backlog ready but big enough that it’s not putting too much stress on us and moving us away from the important tasks.
     3. Even on the weekends we need to communicate as we do when we’re at campus.

Next Steps Assessment

Even with the short amount of time we had on these sprints, we managed to get a very good amount of functionalities and UI screens ready to deploy. One of the first screens and functionality we managed to get working was our log in page. Employees can safely log in or even register for a new account. Which also means that we have user authentication working. We have password encryption working so that passwords are kept safe and secure. We have Employee and Manager permissions so now they see different options on the application depending on their employee level. A manager can view pending employee PTO requests as well as employees can create PTO requests. HR or Managers can filter pending requests by their name, date, and what type of request it is. HR/Managers can also mark requests as approved and will close the request if approved and remove it from pending requests.

However even with this much done we also have functionalities that we could not get finished on time. As of now an employee can’t view their PTO history, employees can’t view the response their managers have left on their PTO request. A manager currently can’t request more information or deny a request. Manager view cannot show currently granted PTO for a specific day. Then finally we have no functionality for a director view. For all of these unmade or unfinished functionalities we would just need more time honestly. Our third and last sprint was sadly a problem because of how our PHP server was just not responding to us for an entire weekend, which we then got working, only for it to then not respond again. So, a lot of time where we could’ve been working on CSS or the backend, was a lot of time put into working on fixing our server problems.