Sprint Planning 11: 3/08 - 3/17

Scrum Master: Sarah S.

Product Backlog

Sprint Planning:

- Sprint Cycle Interval: 03/08/24 - 03/17/24

- Team Capacity: 81 hours
  - Sarah S: 16 hours
  - Carine: 17 hours
  - Brandon: 17 hours
  - Sarah P: 16 hours
  - Kay: 15 hours

- Identify Work:
  - DAR Report: Geolocation API (Original Hours - 5 hours)
  - Peer Review - Space Manager (Original Hours - 3 hours)
  - Peer Review - Space Booking Center (Original Hours - 3 hours)
  - Peer Review - Reservation Confirmation Email (Original Hours - 3 hours)
  - Peer Review - Personal Overview Center (Original Hours - 3 hours)
  - Peer Review - Waitlist (Original Hours - 3 hours)
  - Code - Account Management Center: Employee Account (Original Hours - 4 hours)
  - Code - Space Manager: Creator, List, Modify (Original Hours - 25 hours)
  - Code - Space Booking Center: Availability Display, Reservation, Manage (Original Hours - 17 hours)
  - Unit Test - User Security: Authentication, Authorization (Original Hours - 5 hours)
  - Sprint 11 Documentation (Original Hours - 2 hours)

- Assign Work:
    - Kay
        i. Peer Review - Space Booking Center (Original Hours - 2 hours)
        ii. Code - Space Manager: Creator, List, Modify (Original Hours - 25 hours)
    - Carine
        i. Peer Review - Space Manager (Original Hours - 2 hours)
        ii. Code - Space Booking Center: Availability Display, Reservation, Manage (Original Hours - 17 hours)
    - Sarah P.
        i. Peer Review - Reservation Confirmation Email (Original Hours - 3 hours)
        ii. Code - Account Management Center: Employee Account (Original Hours - 4 hours)
        iii. **Secondary** Unit Test - User Security: Authentication, Authorization
    - Brandon
        i. DAR Report: Geolocation API (Original Hours - 5 hours)
        ii. Peer Review - Waitlist (Original Hours - 3 hours)
    - Sarah S.
        i. Unit Test - User Security: Authentication, Authorization (Original Hours - 5 hours)
        ii. Peer Review - Personal Overview Center (Original Hours - 3 hours)
        iii. Sprint 11 Documentation (Original Hours - 2 hours)

- Task Breakdown:
    - Kay
        - Code - Space Manager: Creator, List, Modify (25 hours)
            - General: Construct tables in database (1 hour)

        - Creator: As a Company/Facility Manager, I want to be able to upload images of my offices' floor plans to the platform, so that I can provide clear spatial layouts to employees (10 hours)
            1. Research and Setup AWS S3 Buckets
            2. Create Models for service
            3. Create upload image service
            4. Construct frontend for SpaceManager View: uploading image form
            5. Connecting ajax with controller
        - List: As a Company/Facility Manager, I want to be able to create a list of reservable spaces that will be located below the floor plan image, so that users can easily identify, select, and reserve office spaces. (8 hours)
            1. Create Models for service
            2. Create space listing service
            3. Construct frontend for SpaceManager View: uploading list of spaces form
            4. Connecting ajax with controller
        - Modify: As a Company/Facility Manager, I want to be able to modify the list of reservable spaces that will be located below the floor plan image, so that I can ensure the list remains accurate and up-to-date (6 hours)
            1. Create modify spaces service
            2. Research how to edit images on S3 buckets
            3. Construct frontend for SpaceManager View: modify spaces form
            4. Connecting ajax with controller
    - Peer Review - Space Booking Center (3 hours)
        - Review design (1 hour)
            1. Identify major positives and negatives

2. Identify unmet requirements
    - Peer Review Document (2 hours)
        1. Explain design qualities and requirements
        2. Share design and test recommendations

    Total: 1 hr + 10 hrs + 8 hrs + 6 hrs + 3 hrs = 28 hrs
    Calculations: 28 hrs - 15 hrs = 13 EP
    Net EP: +13

- Carine
    - Peer Review - Space Booking Center (3 hours)
        - Review design (1 hour)
            3. Identify major positives and negatives
            4. Identify unmet requirements
        - Peer Review Document (2 hours)
            3. Explain design qualities and requirements
            4. Share design and test recommendations
    - Code - Space Booking Center: Availability Display, Reservation, Manage (17 hours)
        1. Finalize database Design (1 hour)
            - Coordinate database table with kay
        2. Availability Display: As an authenticated user, I want to see visual indicators on the view of spaces that clearly distinguish between available and occupied spaces so that it is easier to find an available space to reserve. (4 hours)
            a. Create View - list spaces & color code
            b. Code Display Service
            c. Create unit tests for Display Service**
            d. Create controller
            e. Connect front and backend with ajax
        3. Reserve space: As an authenticated

user, I want the ability to reserve a specific space from the list of available spaces showcased on the view, ensuring I have access to the space when needed. (4 hours)

1. Create View
    a. Available facilities view
    b. Facility info card (floor plan, add availability display) + reservation form
    c. Error/ success pop ups
2. Reservation Service
3. Reservation form
4. Reservation unit tests **
5. Create controller
6. Connect front and backend with ajax

4. User Reservations Display: (4 hours)

1. Create View for list of Current Reservations for that user
2. Create service that gets reservations for that user
3. Create controller
4. Connect front and backend with ajax

5. Cancel Reservation: As an authenticated user, I want the ability to cancel my existing reservations at a facility, ensuring I can cancel before my reservation time based on changing needs. (2 hours)

1. Create Cancel Reservation form
2. Cancel Reservation service
3. Cancel Reservation Service Unit Tests**
4. Create controller
5. Connect front and backend with ajax

6. Modify Reservation: As an authenticated user, I want the ability to modify my existing space reservations at a facility, ensuring I can adjust my

plans before my reservation time based on changing needs. (2 hours)

1. Create Modify Reservation form
2. Modify Reservation Service
3. Modify Reservation Service Unit Tests**
4. Create controller
5. Connect front and backend with ajax

** = if time allows

Total: 3 hrs + 1 hr + 4 hrs + 4 hrs + 4 hrs + 2 hrs + 2 hrs = 20 hrs
Calculations: 20 hrs - 17 hrs = 3 EP
Net EP: +3

- Sarah P.
  - Peer Review - Reservation Confirmation Email (3 hours)
    - Review design (1 hour)
      1. Identify major positives and negatives
      2. Identify unmet requirements
    - Peer Review Document (2 hours)
      1. Explain design qualities and requirements
      2. Share design and test recommendations
  - Code - Account Management Center: Employee Account (4 hours)
    - Research email implementation and design email templates (1 hour)
    - Integrate into project (2 hour)
      - Integrate email service provider
      - Implement functions for different email types
    - Testing (1 hour)
  - **Secondary** Unit Test - User Security: Authentication (SPLIT HOURS - 2 hours)

        - Testing frontend to backend connection (1 hours)
            - Testing if JWT access token is being sent
        - Frontend Testing (1 hour)
            - Implement new frontend testing techniques from lecture to prepare for auto testing

        Total: 3 hrs + 4 hrs + 2 hrs = 9 hrs
        Calculations: 9 hrs - 16 hrs = -7 EP
        Net EP: -7
- Brandon
    - DAR Report: Geocoder API (5 Hours)
        - Coding Tests for each Metric to evaluate each Geocoding API (4 hours)
            - Google API's Response Time of Batch Request of 100 Requests
                1. Code
                2. Testing
            - Google API's Percentage of Successfully Geocoded Addresses Within A Batch Request of 100 addresses
                1. Code
                2. Testing
            - Google API's Maximum Number of Input Geocoded Addresses In A Single Request
                1. Code
                2. Testing
            - Here and Search API's Result Response Time Is Less Than 3 Seconds
                1. Code
                2. Testing

- Here and Search API’s Precision of Coordinates By Precision Points
    1. Code
    2. Testing
- Here and Search API’s Response Time of Batch Request of 100 Requests
    1. Code
    2. Testing
- Here and Search API’s Percentage of Successfully Geocoded Addresses Within A Batch Request of 100 addresses
    1. Code
    2. Testing
- Here and Search API’s Maximum Number of Input Geocoded Addresses In A Single Request
    1. Code
    2. Testing
- OpenCage API’s Result Response Time Is Less Than 3 Seconds
    1. Code
    2. Testing
- OpenCage API’s Precision of Coordinates By Precision Points
    1. Code
    2. Testing
- OpenCage API’s Response Time of Batch Request of 100 Requests
    1. Code
    2. Testing
- OpenCage API’s Percentage of Successfully Geocoded Addresses Within A Batch Request of 100 addresses

1. Code
2. Testing

- OpenCage API's Maximum Number of Input Geocoded Addresses In A Single Request
  1. Code
  2. Testing

- Writing Report (1 Hour)
  - Introduction Section
    - Introducing the APIs
  - Evaluation Section
    - Evaluation results from everyone that leads to our choice of API
  - Data Section
    - Data results that contributed to our choice

- Peer Review: Waitlist (3 Hours)
  - Review design (1 hour)
    1. Identify major positives and negatives
    2. Identify unmet requirements
  - Peer Review Document (2 hours)
    1. Explain design qualities and requirements
    2. Share design and test recommendations

  Total: 5 hrs + 3 hrs = 8 hrs
  Calculations: 8 hrs - 17 hrs = - 9 EP
  Net EP: - 9

- Sarah S.
  - Unit Test - User Security: Authorization (SPLIT HOURS - 3 hours)
    - Testing frontend to backend connection (2 hours)
      - Testing AuthN to AuthZ once a user is logged in
      - Testing JWT access token for claims

        - Frontend Testing (1 hour)
            - Implement new frontend testing techniques from lecture to prepare for auto testing
    - Peer Review - Personal Overview (3 hours)
        - Review design (1 hour)
            1. Identify major positives and negatives
            2. Identify unmet requirements
        - Peer Review Document (2 hours)
            3. Explain design qualities and requirements
            4. Share design and test recommendations
    - Sprint 11 Documentation (2 hours)
        - Planning (1 hour)
        - Stand Ups (25 mins)
        - Retrospective (25 mins)
        - Burndown Charts (25 mins)
        - Backlog (25 mins)

        Total: 3 hrs + 3 hrs + 2 hrs = 8 hrs
        Calculations: 8 hrs - 16 hrs = -8 EP
        Net EP: -8

- <u>Analyze (Team Velocity):</u>
    - Team Capacity: 81 hours
    - Work Item Time: 73 hours
    - Negative EP: Sarah P., Brandon, Sarah S.
    - Positive EP: Kay, Carine
    - Kay and Carine must prioritize peer review in order to start coding for app-specific features
        i. First code review on 3/25
    - Negative EP will go towards extra lecture assignments/implementations, frontend work, and testing
    - Demo Day: Mon 03/18, after Code Reviews