

# Palmyra Racing Association Club Manager: Project Plan

RIT Software Engineering Senior Project 2021-2022

Team Squeaky Training Wheels

Bri Fehring, Jacob Jirinec, Eli Parrish, Matt Robison, CJ Schaefer

## Project Goals and Scope Statement

Replace PRA Club Manager and renewal forms with a new consolidated application that captures existing functionality as well as some new functionality. Create a REST API backend that is easily maintained and expanded on. The backend will store data in a relational database. The backend's functionality will be utilized by the frontend applications, providing users with an intuitive interface for working with the data. In addition to a web app, create a mobile app for both iOS and Android that captures all the same functionality as the web app.

## Team

### **Project Manager** - CJ Schaefer

- Oversees the operation of the team and project as a whole
- Ensures organization and proper maintenance of backlog with Process Manager
- Leads meetings
- Facilitates standups

### **Communications Lead & Website Maintainer** - Eli Parrish

- Manages flow of communication between the team as a whole and the Stakeholder
- Sends the weekly agenda to Stakeholder before our weekly meeting
- Keeps Stakeholders informed on current state of development progress
- Updates the team website with the newest versions of our time tracking documentation, metrics documentation, and artifacts.

### **Process Manager & Automation/Deployment Lead** - Matt Robison

- Ensures organization and proper maintenance of backlog with Project Manager
- Collects and maintains process metrics
- Ensures that work is being done in a timely manner
- Leads the development of the CI/CD pipeline

### **Backend Tech Lead & Secretary** - Jacob Jirinec

- Responsible for helping team members with technical aspects of the backend
- Responsible for researching solutions and technologies to aid in backend development
- Takes Notes during Sponsor Meetings and Team Meetings

### **Web Tech Lead & Mobile Tech Lead** - Bri Fehring

- Provides input regarding design decisions regarding the front-end of the web and mobile applications
- Performs research as needed regarding technologies for the front-end
- Answers questions about and oversees front-end development

## Development Methodology

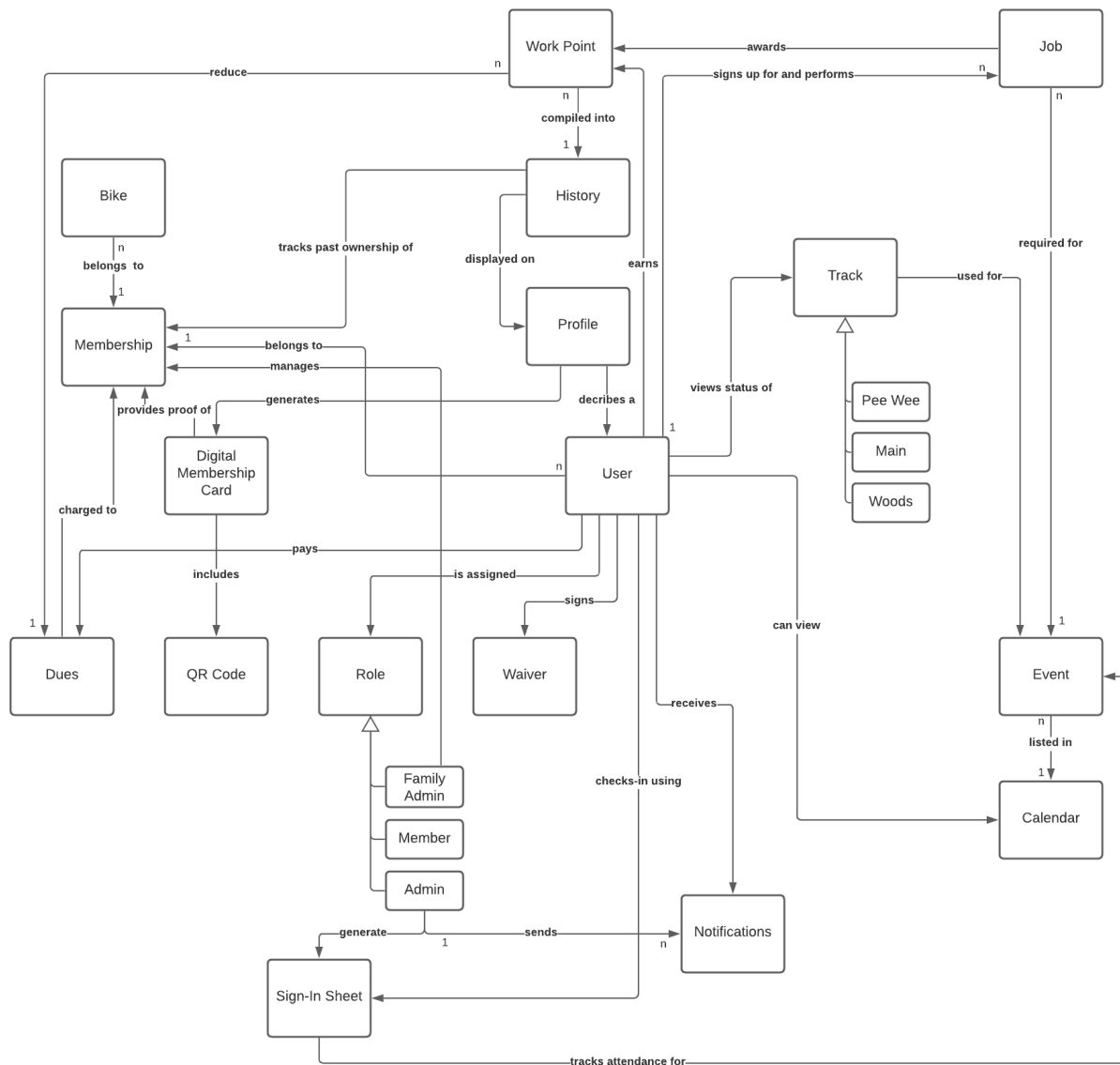
- Planned Activities and Artifacts
  - Scrum
  - Sprints start/end on Tuesdays (2 week duration)
  - Demos to project sponsor on every other Thursday (following the conclusion of a sprint)
    - Will not demo the entire application each sprint - only what's changed and related elements
  - Product Backlog should remain ordered by priority
    - Get priority feedback from sponsor each week
  - Sprint Retrospectives after every sprint
    - Calculate velocity for previous sprint
      - Use to calculate rolling velocity from previous 3 sprints
    - Reflect on progress we have made
    - Open discussion about process
      - What went well?
      - What can be better?
    - Perform additional risk analysis
  - Sprint Planning before every sprint
    - Assign points to any un-pointed stories near top of backlog
      - 8 point maximum
      - Fibonacci sequence for point values (1, 2, 3, 5, 8)
        - 8 is an indication that the story may be too much, look into breaking it up
      - Use traditional planning poker to assign points
        - In case of extreme estimates, have a discussion regarding the ticket in hopes of gaining an understanding of the potential risks and mitigating them
    - Select stories from backlog to work on in next sprint
      - Amount of work taken on will be based on velocity
    - Determine initial assignees for each task
  - Standups 3 times per week
    - In person Tuesdays and Thursday before sponsor meeting
    - Virtual Sundays (5pm)
  - Tuesday meeting slot
    - Standups every other week (mid sprint)
    - Sprint Retrospective and Sprint Planning at start and end of sprints
      - Plan Thursday's demo
    - Create 4-Up
    - Create Thursday agenda
  - Story Card Contents
    - Description
    - Story Points

- Acceptance Criteria
    - These should form the basis of tests for the feature
  - Artifacts
    - Burndown chart
    - Backlog
    - Stories
  - May add research spikes as necessary
- Roles
  - Project Manager (CJ)
    - Ensures organization of backlog
    - Facilitates Standups
    - Leads Planning and Retrospective
  - Process Manager (Matt)
    - Ensures organization of backlog
    - Collects and maintains process metrics
    - Ensures that standups do not run long
    - Follows up with leads to ensure that work is getting done
  - Website Maintainer (Eli)
    - Ensures that metrics, time tracking, and artifacts are published weekly to the website
- Standards and Quality Practices
  - ESLint for code style standards
    - May not have the same .eslintrc for both frontend and backend, but frontend should only modify for specific frontend reasons
  - Automated testing as part of the PR and review process
    - Exact testing libraries TBD
    - Planning a spike for Late Sprint 0/Early Sprint 1 to make these determinations
  - Usability Testing for UI
  - Color Palette for UI
- Tools
  - GitHub for issue tracking and backlog management
  - Google Drive and Discord for remote collaboration
- Metrics and Measurements
  - Velocity
    - Rolling average across previous 3 sprints
  - Completed story points over time throughout the sprint
    - Collected at each standup
  - Time spent in meetings vs. time spent in development

## Domain Model

Link to our LucidChart document:

[https://lucid.app/lucidchart/invitations/accept/inv\\_4407083f-d04e-444c-8b02-04669a1bce0d?viewport\\_loc=-57%2C585%2C2560%2C1232%2C0\\_0](https://lucid.app/lucidchart/invitations/accept/inv_4407083f-d04e-444c-8b02-04669a1bce0d?viewport_loc=-57%2C585%2C2560%2C1232%2C0_0)



## Metrics

- Velocity
  - Captured every sprint, additionally captured as a rolling average for the past 3 sprints
- Earned Value Chart
  - Work with sponsor to assign value to each story
    - Higher value generally equates to higher importance and priority
  - Value should be separate for frontend and backend
- Burndown Chart
  - Updated at each standup
- Code Coverage
  - Calculate separately for both frontend and backend
  - Goal: > 95%
- Usability Testing Metrics
  - Specific to tasks performed, but can include number of clicks to perform a task, number of times assistance is needed to perform a task, and number of errors made during testing.
  - Will be gathered during usability testing sessions with stakeholder(s)

## Communications Plan

The team manages a Discord server, and the sponsor has joined it as well. We will use the server for all of our communications, including sharing meeting agendas and asking questions as necessary. This communication method allows more real-time interactions with the sponsor, which will reduce time spent waiting for answers to questions that are blocking progress. All communication will be handled chiefly through our Communications Lead.