# Trojan Horses Contract

First and foremost, we are here to learn. Don't say that you're "bad" at coding or web development when you've just started, and never count yourself out of a task! Most of us are beginners here; do your best to contribute.

## Goals & Outcomes

We intend to complete our functioning Minimum Viable Product, with all the necessary artifacts associated (the command pipelines, testing, demos, etc) while paying attention to truly applying the software tools and techniques we're exposed to in class.

So by the end of the next 4-5 sprints, we'll have not just the bare MVP, but one that's tested with real usesrs and well-performing as measured by accessibility and performance metrics.

## Success Metrics

We will focus on accessibility and performance with automated metrics, and consider the other -ilities during design and testing.

This project will feel successful if we are able to get feedback from at least 10 users outside of our team on our complete MVP: other CSE 110 students, TAs, and friends.

## Roles

We will fluidly move people across roles with cross training, and based on task needs: there will be times where everyone will take on implementation tasks.

- Glue person
- Design
- Development
- DevOps

## Expectations

1. Whenever you right any code, follow coding standards thoroughly and always get it reviewed

2. Read and respond to communications from teammates thoroughly, in a timely manner (at most 48 hours)
3. Communicate honestly - it's crucial for the team to know the truth
    - if you couldn't get a task assigned to you done for whatever reason, say so, and give a plan for how you will do it, or ask for help
    - share blockers!
4. Engaged participation in our work is a requirement!
    - Support team participation in class: everyone should raise their hands if one person does
    - Group office hour visits
    - Engage with TA and lead messages in Slack
5. Share ideas and feedback
    - we welcome constructive criticism here, and new ideas for improvement!
6. Be accountable for your quality of work and participation
    - Remember that there should be no expectation that teammates will "cover" for you
    - communicate your availability and absences early and often
    - take initiative to follow up if you miss meetings or information

# Project Roadmap

- **Sprint 1: Setup**
    - **Sunday, May 11 -> Design & Setup Phase**
        - DevOps pipeline set up, coding style guide set, tool choices for testing and code coherence set
- **Sprint 2: Design & Dev**
    - **Tuesday, May 13** Design decisions: interfaces update and feedback session after class
        - confirm minimum starting set of features
        - DevOps team will create issues from these
    - **Thursday, May 17** Standup: development progress check
        - team works on feature implementation from designs
        - if design revisits must be made, discuss wit team and make edits
- **Sprint 3: Development**
    - **Sunday, May 20** Standup (remote): progress check with development
        - code review and cross tests
        - if issues closed, support others with blockers or self assign from backlog
    - **Tuesday, May 22** Standup: development progress check
        - code review
        - Thursday is last code review: finish implementation work before then
    - **Thursday, May 24** Standup
        - last code review, hands off for feature implementation
        - come up with plan and timeline for user testing
- **Sprint 4-5: User Testing & Iteration**

    - test with users and other teams
    - get feedback from professor and TAs
    - if time (and backlog is drawn out), brainstorm and upload features
    - fine tune UI
    - tweak design based on feedback
    - optimize
- **Submit project Friday Week 10, June 6**

I understand and agree to the expectations listed, and I will commit to our team outcomes to the best of my ability.

Ayaka Kawano
_____