Table 1: Revision History

Date	**Developer(s)**	**Change**
2022-09-26	Jonathan Cels	Team meeting and communication plans, personal role, workflow plan, coding standard, project scheduling, started technology section
22-09-26	Alexander Van Kralingen	Introduce CI/CD plan, team member role
22-09-26	Arshdeep Aujla	Introduce Hardware List 6, 3.2 Add Roles
...	...	...

Development Plan
ProgName

Team #, Team Name
Student 1 name and macid
Student 2 name and macid
Student 3 name and macid
Student 4 name and macid

[Put your introductory blurb here. —SS]

1 Team Meeting Plan

The team will meet weekly on Thursdays at 10:20 AM until 11:20 AM. Team members are expected to attend the Thursday lecture and meet up following the lecture. On the event that no lecture is scheduled, the meeting time shall be changed to 9:30 until 11:20AM.

Additional meetings will be scheduled when necessary using the communication methods outlined in the Team Communication Plan.

2 Team Communication Plan

The team will communicate over a Discord group channel. Each team member is expected to have Discord downloaded and readily accessible on at least one of their devices. In the case of emergencies or time-critical situations, a team member will use SMS text messaging in place of Discord. Text messages will only be used in critical situations.

Issues will be managed using a kanban board, implemented with a Git project board. Tasks will be split into small, workable items with differing deadlines and importance.

3 Team Member Roles

3.1 Alexander Van Kralingen

- Embedded systems software development

- Continuous Integration/Deployment management
- Assisting with development of web application
- Code reviewing for both embedded and web application code
- Unit/integration testing development

3.2 Arshdeep Aujla

- Assisting in building the chess board and customise the chess pieces
- Installing the hardware components on a breadboard
- Soldering components on the PCB
- Mapping the relation between the inputs and outputs using Karnaugh Map and State Machine Table

3.3 Jonathan Cels

- Enabling the Bluetooth connection between the application and the microcontroller
- Leading testing initiatives and ensuring that thorough testing is completed
- Assisting with development of web application
- Editing and formatting of all documentation before submission

3.4 Joshua Chapman

3.5 Rupinder Nagra

4 Workflow Plan

The project will use the feature-branch methodology in Git. An outline of the workflow is as follows:

1. Pull any changes from the master branch
2. Create a new branch for development of a specific feature or subsystem
   (a) Use branch names that are descriptive to the feature
3. Commit code frequently with descriptive messages
4. Add unit and integration tests for the changes
5. Push code to branch
6. Create a pull request

7. Another team member reviews and approves or rejects the pull request
   (a) The tests are reviewed and more tests are created if necessary
   (b) The pull request cannot be approved if all tests do not pass
8. Merge feature branch into the main branch

5 Proof of Concept Demonstration Plan

What is the main risk, or risks, for the success of your project? What will you demonstrate during your proof of concept demonstration to convince yourself that you will be able to overcome this risk?

6 Technology

- Languages and Frameworks
  - **JavaScript, HTML, CSS, and React.js:** Frontend web based development
  - **Python, FastAPI, Node.js:** Backend development and bluetooth connection
  - **C:** Microcontroller programming
  - **MongoDB:** Database solution
- Linting
  - **ESLint:** JavaScript development in VSCode
  - **Flake8:** Python development in VSCode
- Testing Frameworks
  - **React Testing Library:** JavaScript testing framework for React.js
  - **PyTest:** Python testing framework
- **Heroku:** Deployment
- **VSCode:** Code editor
- **GitHub:** Version control and project management
- **LaTeX:** Documentation
- Libraries and API's
  - **Bluez, PyBluez:** Enabling bluetooth data transfer
  - **Stockfish.js:** Chess engine to find optimal moves

- **Continuous Integration (CI)** GitHub Actions are to be used for CI, with the following general workflow:
  - Running on ubuntu-latest
  - **Pull Requests (PR):**
    * Requires one reviewer
    * Requires development branch to be up-to-date with main before approval
    * Requires building/linting and tests to run completely without errors
  - **Build Embedded Code:**
    * Triggered on PR from any hardware/* branches
    * Checkout hardware/* branch
    * Build all C code
    * Run all C tests
  - **Build Web App Code:**
    * Triggered on PR from any webapp/* branches
    * Checkout webapp/* branch
    * Lint all Python code
    * Lint all JavaScript code
    * Build JavaScript code
    * Run all JavaScript tests
    * Run all Python tests
  - **Build LaTeX docs:**
    * Triggered on push (any branch) with *.tex file changes
    * Build all latex documents
  - **Embedded Testing:**
    * Triggered on PR with Hardware Label opened, updated or closed
    * Run all C tests
  - **Web-App Testing:**
    * Triggered on PR with Webapp Label opened, updated or closed
    * Run all JavaScript tests
    * Run all Python tests
- **Continuous Deployment (CD)** GitHub integration in Heroku to be used for continuous deployment:
  - **Deploy:**
    * Triggered on push to main
    * Push to main event occurs when merging a PR

    - Web application deployed to Heroku on successfully merged PR.

- **Hardware**
  - **Chess Board:**
    - Chess squares in two different colors
  - **Chess Pieces:**
    - Customised chess pieces with magnets attached at the bottom of each piece for their detection
  - **PCB:**
    - Through hole PCB prototype board
  - **Breadboard:**
    - For prototyping the integration of hardware components
  - **Hall Sensors:**
    - Sensors below the board to detect the different chess pieces
  - **Magnets:**
    - To attach below the chess pieces for their detection on the chessboard
  - **Wires:**
    - To connect sensors to the controllers, power LEDs
  - **Processor:**
    - An 8 Analog Inputs 32 GPIO Bluetooth Module
  - **Power supply cable:**
    - Wall extension cable to the chess board
  - **Power supply circuit**
    - 120V AC to V DC power converter
  - **LCD Display:**
    - Can be a 7-segment display or an LED
  - **Switches:**
    - To toggle between different modes
  - **LEDs:**
    - To light up the squares to display the best moves or to teach the game

7 Coding Standard

The project will follow the Airbnb style guide for JavaScript development, and use the Flake8 style guide for Python development.

8 Project Scheduling

The project will use a GitHub project board to track and schedule tasks on a weekly basis.

Technical roles are decided on the basis of prior knowledge and interest. In the case of the team leader, the role will change with every deliverable. Team members are responsible for decomposing their tasks into kanban items, individually or alongside other team members working on the same task.

Major milestones will be tracked on the project board. Milestones include both hard and soft deadlines for task completion. Hard deadlines are the project deliverable due dates. Soft deadlines are decided by the team for the completion of technical tasks.