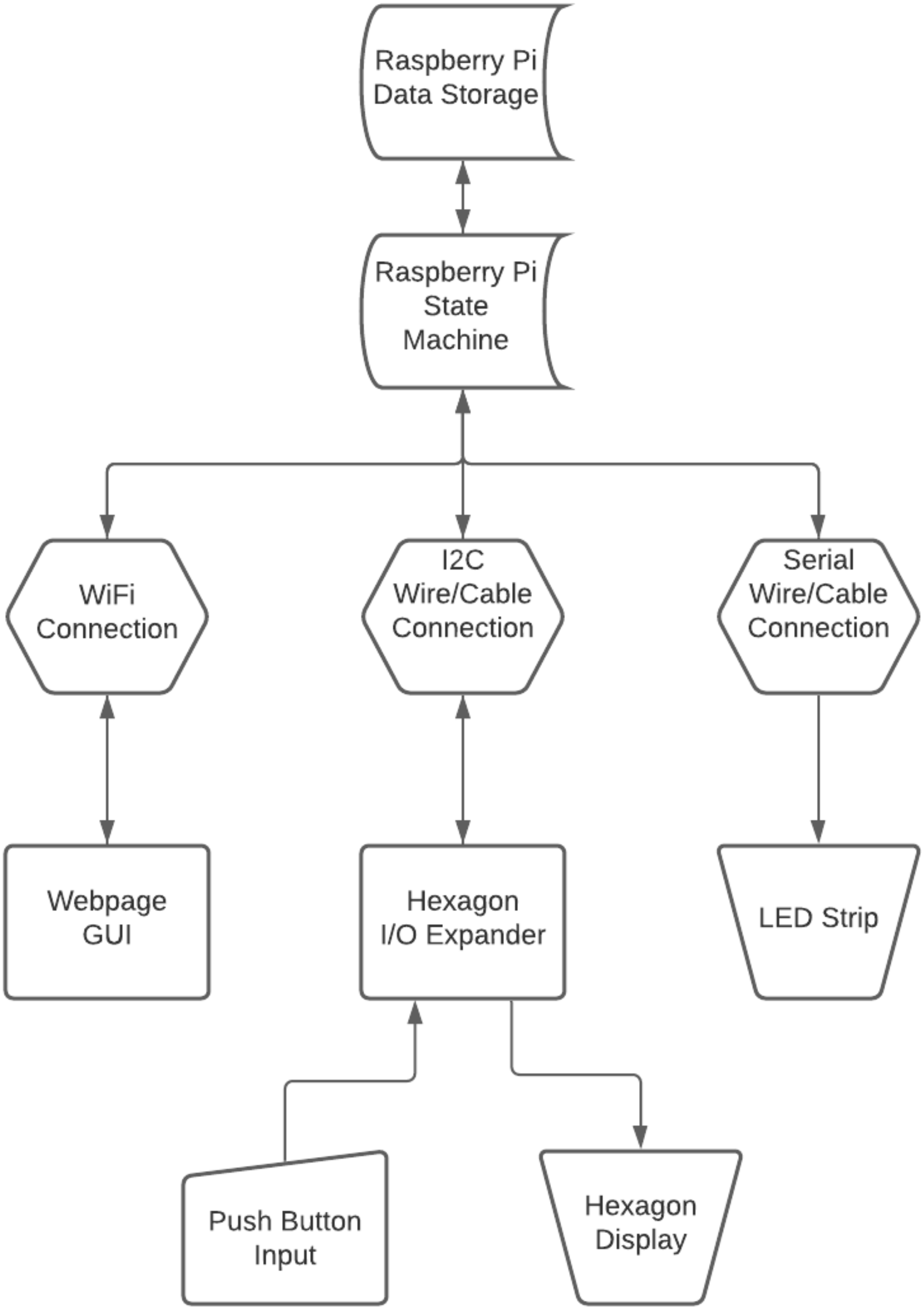
1. Provide a document with a system block diagram and a list of all major subsystems.


Raspberry Pi Data Storage
Raspberry Pi State Machine
WiFi Connection
I2C Wire/Cable Connection
Serial Wire/Cable Connection
Webpage GUI
Hexagon I/O Expander
LED Strip
Push Button Input
Hexagon Display

a. *Raspberry Pi Data Storage:* The Raspberry Pi shall contain a set of variables that encode the game state and may be accessed by the state machine to determine the appropriate outputs.
b. *Raspberry Pi Game API:* The Raspberry Pi shall control the entire system through an API that is accessible over wifi.
c. *WiFi Connection:* The Raspberry Pi shall host a local webpage that can be accessed from smartphones (or other devices) through a WiFi connection.
d. *I2C and Serial Wire/Cable Connections:* The Raspberry Pi shall be connected to the hexagons and LED strip through wires and cables that will carry I2C and serial communications.
e. *Webpage GUI:* There should be software that will use a GUI to take user input for more complex actions and communicate game information to the players that is not conveyed with the board itself.
f. *Hexagon I/O Expander:* Individual chips will be used for each hexagon to delegate the inputs and outputs of individual board spaces.
g. *Push Button Input:* There will be push buttons on the board that allow players to select specific "nodes" on the board as inputs during certain game actions.
h. *Hexagon Display:* The hexagons will also display information on each tile, such as the resource type and number on which it produces resources.
i. *LED Strip:* There will be an addressable LED strip that will display the locations of various "developments," such as settlements and roads.

2. Provide a list of requirements for each subsystem.

Subsystem and Interface Requirements:

Raspberry Pi Data Storage

The data storage system within the Raspberry Pi must meet the following requirements:

- The data system must be completely encapsulated in software
- The data shall not exceed the memory requirements of the Raspberry Pi
- The state machine must have both read and write access to the data storage system
- The data system should virtually contain all of the information on the board, including:
    - The resource types and numbers of each tile
    - The location and owner of each settlement, city, and road
    - The location of the robber
- The data system should virtually contain the resources players control, including development cards

Raspberry Pi State Machine

The state machine within the Raspberry Pi must meet the following requirements:

- The state machine must be completely encapsulated in software
- The state machine's script shall not exceed the memory requirements of the Raspberry Pi

- The state machine must have read and write access to the data storage
- The state machine must be able to generate a local webpage
- The state machine must have access to the WiFi, I2C, and serial communications
- The state machine must have the following states:
    - Rolling dice
    - Moving the robber
    - Selecting game actions for turn (i.e. trade, build, pass turn)
    - Proposing trades
    - Accepting trades
    - Selecting development to purchase
    - Selecting location for developments

WiFi Connection

The WiFi connection must meet the following requirements:

- The connection must meet FCC requirements
- The connection must have a range of at least 10 feet
- The connection must be accessible from smart phones, tablets, and laptops
- The connection must support multiple devices at once

I2C and Serial Wire/Cable Connections

The I2C and serial connections must meet the following requirements:

- Connections to the hexagons must be capable of reading push button inputs to the Raspberry Pi
- Connections to the hexagons must be capable of writing initial tile information (such as resource type and number) to the hexagon processor
- Connections to the hexagons must be capable of writing when the robber moves
- Connections to the LEDs must be able to address the location and owner of new developments (such as cities, settlements, and roads)

Webpage GUI

The webpage GUI should meet the following requirements:

- There should be separate GUIs for each player's device in order to distinguish which player is using which device
- The GUI should be friendly to multiple types of devices, including smartphones, tablets, and laptops
- Users should be able to input the following game actions:
    - Roll dice
    - "Maritime trade" with the bank
    - Propose trade to another player
    - Accept trade from another player
    - Build a development
    - Select a development type to build

    - Play a development card
    - Pass turn
- The GUI should display the following information for each player:
    - Number and type of resources owned
    - Development cards owned
    - Who possesses Longest Road and Largest Army
    - Running Victory Point total of player
    - Known Victory Point total of opponents
    - Explanation of mistakes or illegal moves that the player attempts

Hexagon I/O Expanders

The hexagon I/O expanders must contain enough I/O pins for the following interfaces:

- I2C communication with the Raspberry Pi
- Display of hexagon's resource type
- 7-segment LCD display
- Display of robber's presence
- Input from push buttons

Push Button Input

The push button input must allow players to select the following locations during associated actions:

- A unique hexagon edge when performing the Build a Road action
- A unique hexagon corner when performing the Build a Settlement or Build a City action
- A unique tile when relocating the robber

Hexagon Display

The hexagon display must convey the following information:

- The resource type produced by the tile
- The number on which the tile will produce resources
- The presence or absence of the robber on the tile

LED Strip

The LED strip must meet the following requirements:

- The LED strip must be addressable so that individual LEDs may be illuminated with different values
- The LED strip must be capable of displaying at least four distinct colors, one for each player, plus an "off" state for edges and corners not controlled by any player
- The LED strip must be present at each edge and corner of the board that a player may control

3. Give a written plan for achieving the first design review.
    a. Within the first week or two, we’ll achieve points 1 and 2 through group Zoom calls and potential independent work. We have some subsystems’ detailed designs already completed. The rest of the subsystems will be characterized during that time.
    b. Then, the following week or two, we will consult the datasheets and our detailed designs in order to figure out the essential connections for all of our major components. This would require us to have a complete list of the actual parts that we’re going to be using and how they’re intended on being connected. Once we’ve done this, we can calculate current and voltage requirements. Also, we will finalize all communication protocols so we can determine the correct number of pins and buses needed in our design.
    c. For any potential unclear problems, we have documented some problems thus far, and will encounter further questions as we select our components and make our detailed designs.
    d. Once everything has been completed, we will be able to upload our Proposal and High Level design documents on the team web site with links, by following the instructions in the “Webinfo.pdf” document.