Engineering Art

DESIGN DOCUMENT

sdmay24-17
Client - Dr. Rachel Shannon
Faculty Advisor - Dr. Rachel Shannon
Karandeep Sandhu - Electrical Engineering
Alexandra Whitmer - Software Engineering
Austin Krekula - Cyber Security Engineering
Seyedehbahareh Hashemimovahed- Computer Engineering
Grant DeWaay - Software Engineering / Interdisciplinary Design

Executive Summary

Development Standards & Practices Used

- Hardware Practices:
  - Meticulous design documentation
  - Comply with industry standards and regulations
  - Compatibility and integration testing
  - Prototyping and iteration
- Software Practice:
  - Version control and code maintenance
  - Code review
  - Security and vulnerability management
  - Documentation and User Guide
- Art Practice:
  - Artistic style and expression
  - Creative development and conceptualization
  - Fusion of art and technology
  - Narrative and emotional expression
- IEEE Standard:
  - IEEE P2784：The Guide for the Technology Evaluation of Smart City Projects.
  - IEEE 1680 Series: Standards for Environmental Assessment of Electronic Products. Design principles for sustainable and eco-friendly smart city elements in the art piece.
  - IEEE 802.3: Ethernet
  - IEEE 802.11: Wireless LAN (Wi-Fi)
- ISO Standards:

- ISO 9075-13:2016 Information technology -- Database languages -- SQL -- Part 13: SQL Routines and types using the Java TM programming language (SQL/JRT)
- ISO 14882:2020 Information technology - Programming Languages - C++
- ISO 23270:2018 Information technology - Programming languages - C
- ISO/IEC 27001: Provides a systematic approach to managing sensitive company information so that it remains secure. It encompasses people, processes, and IT systems, assessing and mitigating the risks to which they are exposed

Summary of Requirements

- Research and explore the 14 National Academy of Engineering Grand Challenges for Engineering
- Choose several of the challenges to explore further
- Narrow down our options to just one challenge to tackle
- Brainstorm solutions to the redefined problem
- Create a prototype

Applicable Courses from Iowa State University Curriculum

COM S 227, 228

SE 309, 421

DES 150x, 230

List of figures & tables

Figures

Tables

1 Team

1.1 Team Members

Karandeep Sandhu

Alexandra Whitmer

Austin Krekula

Seyedehbahareh Hashemimovahed

Grant DeWaay

Zheyuan Zhang

1.2 Required Skill Sets for Our Project

- Programming language experience
  - We will need Python experience for Front-end and Back-end development.
- Computer Graphics experience
  - Using computer graphics, we will create 2-D or 3-D models and designs
- 3-D printing experience
  - Required for printing the grid spaces
- Testing skills (Software/Hardware)
  - Based on the requirements of the project we will be conducting different stages of testing such as unit testing, integration, etc. to ensure the correctness
- Censor/Signal Processing
  - We will use this skillset for data transfer. The data pulled from the grid to PuTTY and vice versa (hardware and software communication) into readable values.

1.3 Skill Sets covered by the Team

Zheyuan: Matlab, AutoCAD and some simple C language.

Alexandra: Java, frontend development, and some hardware experience.

Karandeep: Circuit design, Matlab, Cadence Virtuoso, and some Verilog and simple C.

Grant DeWaay: Wide array of programming languages along with modeling tools like Solidworks and Grasshopper 3D and manufacturing techniques like 3D printing, sewing, embroidery, and laser cutting. Experience with double diamond design model and design thinking.

Austin Krekula: Security oriented design thinking, risk management framework, information security, Linux OS security, Windows OS security, networking security, applications of cryptography, discrete mathematics, Python, C, Java, Verilog, assembly language (MIPS & x87 instruction sets), software reverse engineering, digital forensics, embedded systems engineering, microprocessor design, digital logic, algorithm analysis.

Bahar: Java, some simple C, some embedded system and VHDL.

1.4 Project Management Style Adopted by the team

We will be using the Agile project management style for the duration of this project in addition to the double diamond design method throughout both semesters of senior design.

1.5 Initial Project Management Roles

Everyone will participate in our team and we all expect that as tasks arise we will distribute them to meet everyone's specific skill set and availability. We have listed everyone's skill sets above. Our client is our advisor so we will meet with them on a weekly basis.

Artistic and creative direction that includes non technical elements will be primarily handled by Grant DeWaay in addition to anyone else on the team that expresses interest in contributing to this aspect of the project. This role may transfer to anyone else in the group if there is overwhelming distaste of the ideas being presented, or the practices being implemented. Elements that fall under this category include the following:

- Shape and form
- Color palette
- Materials used for non technical elements
  - ex: the enclosure
- Presentation
- Messaging
  - NOT including the selection of 1 of the 14 engineering challenges
  - NOT including changing project scope for the sake of greater artistic expression unless agreed upon by the entire group

2 Introduction

2.1 Problem Statement

The problem we need to solve for our design project is to educate the public about the contemporary challenges that engineers tackle. We must convey this in an artistic way that demonstrates the use of electrical, computer, software, and/or cybersecurity engineering in our final product. We are given a list of 14 modern engineering challenges that we can base our art project here: http://www.engineeringchallenges.org/challenges.aspx

2.2 Requirements & Constraints

Our requirements and constraints will be oriented towards improvement of quality of life and environmental sustainability.

Improvement on urban infrastructure

- Functional Requirements (areas that we want to address and improve)
  - Power - Ensure efficient power usage on both the micro and macro levels.
  - Sustainability - Reducing emissions (not increasing emissions) and safety for wildlife/environment.
  - Public Service - Increase public awareness, improve lives by making implementable and cost effective service access.
  - Transportation - Improvement in various modes of transportation (subways, cars, business, etc.).
  - Connectivity - Broadband or other technical communication between elements of our project and the rest of the urban environment.
  - Building infrastructure - Improving infrastructure in terms of efficiency, affordability, longevity, technology, etc.
- Environmental Requirements
  - Reduced emissions, to not increase emissions (constraint).
  - Reduced power consumption (constraint).
  - Safety for wildlife/environment
- Economic
  - Realistic cost guidelines and adherence to those guidelines (constraint).
  - Can reduce operation costs compared to normal cities (constraint).
  - Can have certain industries or sectors that generate profit
- Legal
  - Improvements and changes are implemented in an ethical and just manner
- Maintainability
  - Affordability (constraint).

- Sustainability specifically durability
- Individuals can be trained for the upkeep of the society

- Performance
    - Improvements to cities to properly address problems in Power, Sustainability, Public Service, Transportation, Broadband, Building Infrastructure and make improvements in those areas in an appropriate manner

2.3 Engineering Standards

The standards our project will have are going to come from national, international, and volunteer industry standards organizations such as IEEE, IETF, ANSI. The applicable standards will cover ethernet and wireless communication, computer language use, database use, technological safety, and environmental sustainability.

2.3.1 IEEE Standards:

IEEE 802.3: Ethernet

IEEE 802.11: Wireless LAN (Wi-Fi)

IEEE 802.15.4: Used for low-rate wireless personal area networks (commonly used in IoT applications).

IEEE P2784: The Guide for the Technology Evaluation of Smart City Projects.

IEEE 1680 Series: Standards for Environmental Assessment of Electronic Products. Design principles for sustainable and eco-friendly smart city elements in the art piece.

Adhering to IEEE standards ensures reliable, compatible, and sustainable technological solutions for urban infrastructure.

2.3.2 ANSI Standards:

ISO 9075-13:2016 Information technology -- Database languages -- SQL -- Part 13: SQL Routines and types using the Java TM programming language (SQL/JRT)

ISO 14882:2020 Information technology - Programming Languages - C++

ISO 23270:2018 Information technology - Programming languages - C

ISO/IEC 27001: Provides a systematic approach to managing sensitive company information so that it remains secure. It encompasses people, processes, and IT systems, assessing and mitigating the risks to which they are exposed.

IEC 60950: Safety of information technology equipment: Pertains to the safety of IT equipment. The standard defines the general requirements for the safety of such equipment, including protection from electrical hazards, thermal risks, mechanical hazards, and more.

IEC 61000: Electromagnetic compatibility (EMC): A series of standards addressing electromagnetic compatibility (EMC). It encompasses both electromagnetic interference (EMI) and susceptibility to interference.

By following ANSI standards we guarantee the quality, safety, and efficiency of urban infrastructure products and services.

2.3.3 IETF STANDARDS

IPv6 (RFC 8200): With the increasing number of devices in smart cities, IPv6 offers a vast address space that can accommodate this growth, making it essential for future-proof smart city deployments.

6LoWPAN (RFC 4944, RFC 6282, and more): This standard describes how to efficiently use IPv6 over low-power wireless personal area networks, which is crucial for battery-operated IoT devices prevalent in smart cities.

RPL (RFC 6550): The IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) is crucial for establishing routes in low-power and lossy networks, like those in IoT settings.

2.4 Intended Users and Uses

The general population will benefit from the results of our project since this project is intended to be a display. They will use it by essentially living in these improved urban infrastructure. We plan on sharing this project with the student body of Iowa State so they can learn how urban infrastructure can be redesigned and redeveloped. We will also be able to showcase how this improved infrastructure can be used on the global level.

3 Project Plan

3.1 Task Decomposition

In order to solve the problem at hand, it helps to decompose it into multiple tasks and subtasks and to understand interdependence among tasks. This step might be useful even if we adopt agile methodology. If we are agile, we can also provide a linear progression of completed requirements aligned with our sprints for the entire project. At minimum, this section should have a task dependence graph, description of each task, and a justification of our tasks with respect to our requirements. We may optionally also include sub-tasks.

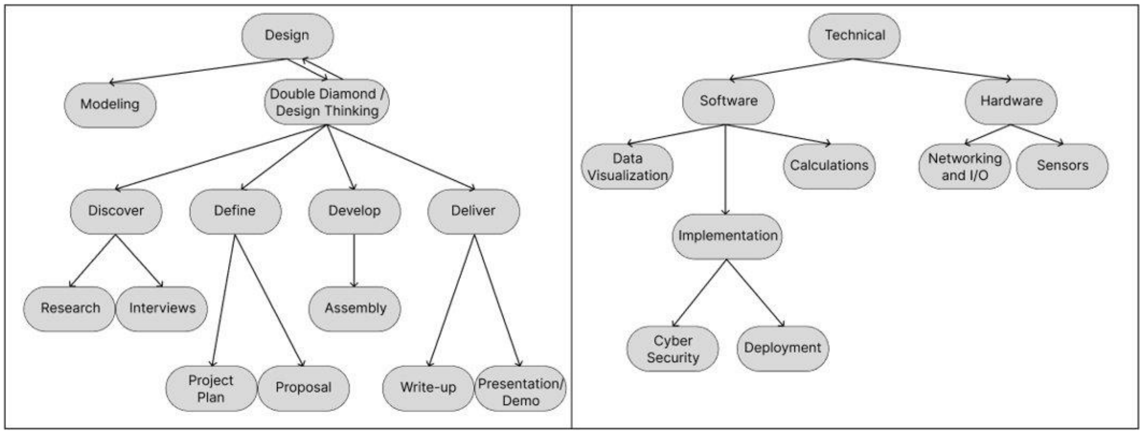

Figure 1: Integrated Design and Technical Development Process Map

- Design - Planning out the requirements that are derived from the project proposal
    - Modeling - Includes any artistic elements that enhance the final deliverable
    - Double Diamond / Design Thinking
        - Discover - Performing research on the topic we are given so we can make informed decisions regarding how we can build a solution to the problem statement
            - Research - Reading text sources that relate to the genre of problem
            - Interviews - Talking with experts on campus to learn more about what they would like to see out of our project
        - Define
            - Project Plan - We now contribute our findings from the Discover phase to culminate a plan that meets the needs and requirements of both the project and the client
            - Proposal - Verify with the client that the team's findings and project plan align with their interests and intent.
        - Develop
            - Assembly - Follow through with our project plan and verify that we are meeting expectations that were set by our teammates, and our client
        - Deliver
            - Write-up - Write thorough documentation that explains all the steps of the process necessary to operate our project

            - Demonstrate - Showcase our project to the client, using technical language that the audience understands

- Technical
    - Software - components that visualizes data from the hardware and calculations made on the backend
        - Data visualization - Show data that is expressive and informative that captures the attention of the target audience
        - Implementation
            - Cybersecurity - Cybersecurity that ensures the project will stay live without any major interruptions and low maintenance
            - Deployment- Deployment that is easy to configure, with simple teardown and setup, even for inexperienced users.
        - Calculations - Calculations that are essential to demonstrate the impacts of designed urban development
    - Hardware - components that capture the user input and enables the installation to be interactive for people of all ages
        - Networking- Networking which communicates the state of the system, whether that be user interaction or other state changes, in a timely manner so the software can perform the tasks needed with the lowest latency
        - Sensors - Sensors that are accurate and are tested and measured to gauge the correct amount of force or interaction before certain data is sent to the software.

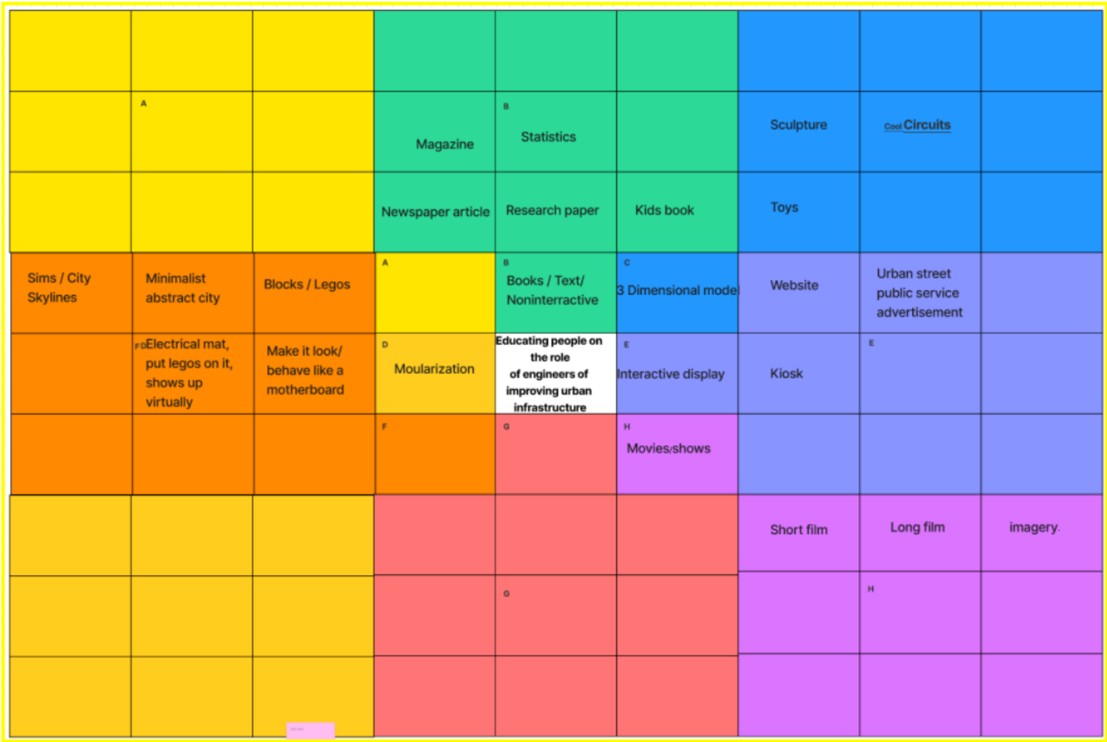

Figure 2: Creative Communication and Media Matrix

3.2 Project Management/Tracking Procedures

We are adopting waterfall management.

At the outset of our project, we referred to the "14 Grand Challenges for engineers" and selected "restore and improve urban infrastructure" as our primary focus. This decision marked the first phase of our project. In alignment with the characteristics of the Waterfall model, once this phase was set, we proceeded to the next step. Thus, our subsequent task was to delve into the specific issues associated with urban infrastructure.

After in-depth discussions, we identified six primary research areas:

-Power Group
-Energy sustainability
-Cybersecurity
-Civil
-Business
-Urban design

As part of the Waterfall model approach, we believe that a comprehensive review of the work from a preceding phase is vital before embarking on the next. This ensures that we don't overlook any crucial information or essential details.

Consequently, in line with the linear progression of the Waterfall model, the objective of our next phase is to interview experts within our campus who have notable research and insights in the aforementioned areas. Through their professional feedback and opinions, we hope to refine our research directions and strategies, laying a solid foundation for the subsequent development and implementation phases.

We will use Discord for quick communication and for short voice calls when there is an action that needs fast input from multiple members of our team. Figma will be used for visual collaboration in addition to idea generation. The instructor-provided GitLab repository will be utilized for version control of our software components with Git. Additionally, we will be keeping track of milestones and task progress using GitLab's built in Issues and Tasks tracker.

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

What are some key milestones in our proposed project? It may be helpful to develop these milestones for each task and subtask from 2.1. How do we measure progress on a given task? These metrics, preferably quantifiable, should be developed for each task. The milestones

should be stated in terms of these metrics: Machine learning algorithm XYZ will classify with 80% accuracy; the pattern recognition logic on FPGA will recognize a pattern every 1 ms (at 1K patterns/sec throughput). ML accuracy target might go up to 90% from 80%.

In an agile development process, these milestones can be refined with successive iterations/sprints (perhaps a subset of our requirements applicable to those sprint).

Our biggest key milestones for this project are coming to a decision on which challenge we want to tackle, deciding how we want to tackle this challenge, and each hardware and software iteration thereafter. Two of these milestones have already been reached by the team, and we are now in the process of creating new iterations of our design and developing them to their extent. We move on to a new iteration when we receive feedback or a team member comes up with a new or improved idea for one of our designs. With our current design, we will be measuring if the photoresistor grid works by having our printed buildings light up when the sensors read a light level below a specific threshold. We will measure that the Arduino is programmed correctly by having our display output color-coded blocks, which should be oriented in the correct direction at least 95% of the time. The physical blocks should match up with the virtual blocks at least 90% of the time, accounting for orientation errors due to the shape of some of our buildings.

3.4 Project Timeline/Schedule

• A realistic, well-planned schedule is an essential component of every well-planned project

• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

• A detailed schedule is needed as a part of the plan:

– Start with a Gantt chart showing the tasks (that we developed in 2.2) and associated subtasks versus the proposed project calendar. The Gantt chart shall be referenced and summarized in the text.

– Annotate the Gantt chart with when each project deliverable will be delivered

• Project schedule/Gantt chart can be adapted to Agile or Waterfall development model. For agile, a sprint schedule with specific technical milestones/requirements/targets will work.

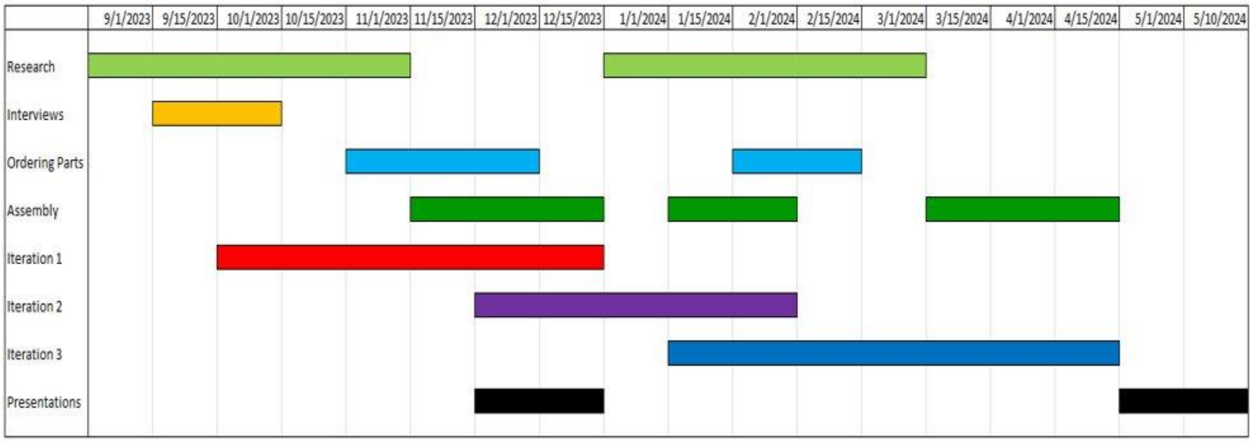

Figure 3: Project Timeline Gantt Chart

We are currently in the Iteration 1 section of our Gantt chart. What we are presenting to the faculty board during our first initial demo will be the beginning stages of our iteration 1 prototype. Using the principles that we acquired from our research, interviews, and ideation, our prototype will reflect some of, although not fully realized ideas that we found.

3.5 Risks And Risk Management/Mitigation

During this early development stage the risks we face are of low to medium impact. Problems that might affect the long term success of our project are generally those that impede planning, clarity of vision, coordination, and execution of tasks.

Such risks are:

- Miscommunication
  - Different vision of project outcome
  - Different understanding of requirements
- Disagreement
  - Different design goals
  - Different ideas about implementation
- Misdirection
  - Planning towards a dead-end
    - Unusable hardware
    - Ill designed software
    - Not seeing solutions
    - Accidental intellectual property theft

These risks are inherent in all team efforts. Our mitigation strategy is built into the waterfall+agile design management methodology. We will be using regular directed meetings and status updates integrated into each sprint to bring up misunderstandings or other issues as early as possible.

The nature of these risks is that they will always be present and no risk management control will be able to completely eliminate them, but the Agile project methodology allows the reassessment of risks and risk mitigation with each sprint, therefore we will be able to regularly reassess these issues.

Later in development we need to consider what kind of materials we can use when constructing the physical aspect of our project. We need to use non conductive materials when building our playmat and tiles being used to prevent endangering the public with electrocution. Everything needs to be safe for the public, or else we are doing everyone a huge diservice

3.6 Personnel Effort Requirements

Member Name	**Task**	**Hours Spent**
Everyone	Research - Everyone is to research areas of our project. • Read 14 Grand Challenges For Engineers and find a topic that we are interested in. • Conduct additional research on those topics of interest. • Select a topic to pursue as a group (we have selected Restore and improve urban infrastructure as our topic). • Conduct research on	• 5-8 hours spent reading documents and conducting research on topics that interest us. • After selecting a topic as a group, spend roughly 5-8 hours researching our topic.

	said topic.	
Everyone (individually)	Faculty meeting - Everyone is to research their specific area and meet with a professor in that area to discuss our project as it pertains to their specific field. • Karandeep - engineering sustainability • Ally - business (recommendations for public services they'd like to see development in) • Zheyuan - Power group • Austin - cybersecurity • Grant - urban design (college of design) • Bahar - civil	Everyone is expected to spend roughly 5-8 hours researching and meeting with their respective professors. The meeting doesn't have to be very long; however it is expected that the group member has conducted sufficient research in the topic to be able to ask the professor/advisor detailed questions.
Everyone (together)	Finalizing the problem - As a group we will meet and discuss the information we have gathered by meeting with our respective faculty members. We will then decide on how we want to proceed moving forward into the Define phase of our project.	1-3 hours
Everyone (together) and for ordering parts we will select	Design brainstorming and ordering parts - We will meet	We will discuss our design over the course of 2-3 weeks

someone during the meeting to do it.	and discuss the design of our project and then begin to order parts. We will select someone during these meetings to conduct research on where these parts can be purchased and order them.	and will meet regularly during those 2-3 weeks and order the parts as needed. Roughly 3 hours per week will be spent weekly on this.
Everyone	Iteration 1 - This will be the design planning section where we have parts ordered and will conduct testing as needed. It will be very modular to ensure that components of our design are coming together and we will constantly be updating and implementing our design during this phase.	As of right now we don't know because we have not reached this phase in our project. When we reach this phase we will update this section however for right now it is expected that all group members try to contribute at least 5-8 hours a week.
Everyone	Iteration 2 (second semester) - This will be a mix of continuously updating our design as we encounter issues and also implementing our design. There haven't been any specific tasks assigned yet for this portion because we have not yet reached this phase in our project. When we reach this point we will update this section to become more specific.	As of right now we don't know because we have not reached this phase in our project. When we reach this phase we will update this section however for right now it is expected that all group members try to contribute at least 5-8 hours a week.

Table 1: Personal Effort

3.7 Other Resource Requirements

Identify the other resources aside from financial (such as parts and materials) required to complete the project.

- **Access to fabrications tools such as 3d printers in the Student Innovation Center and a PCB printer. We may also need to utilize woodworking and other forms of manufacturing.**
- **Input from faculty experts so the project can be designed to be as relevant and useful as possible as an education tool.**
- **A space for our team to assemble, store, and use the components of our project. Our team's project will be relatively large so we most likely need a dedicated studio or a designated space where we can work, store parts, and collaborate in-person.**

4 Design

Generally, the public lacks awareness of engineering problems, specifically the 14 Grand Challenges for Engineering, and their states of development. The understanding that the public has of the Grand Challenges (what they are and current state) influences their enthusiasm and engagement. If they do not understand the Challenges, then they will not have enthusiasm or engagement with their solutions. If they understand them then they will (more likely) have enthusiasm and engagement with their solutions. To address this issue, we aim to develop an interactive installation that appeals to people of all ages and effectively conveys Grand Challenges and their current states of development. An enthused and engaged public is a source of funding, research (new minds) and application. If the public is more educated, then Grand Challenges are more likely to be solved in a faster time. We will use an artistic demonstration to engage and educate the public in the state of urban infrastructure development as a means to ensure they support the research and development of Grand Challenges.

4.1 Design Content

Briefly describe what is the design content in our project.

Our project has many design considerations that cover both hardware and software that involves multiple different disciplines that are necessary to research and be informed about. As our project is an interactive installation, we need to think about how every party will interact with our creation. This not only includes the audience that will be placing blocks and playing with our project, but also those who need to store and maintain it. As such, we have to ask questions about how quickly it can be assembled and disassembled, and how it can be properly stored so non-technical faculty can keep educating the public about engineering challenges years after we graduate.

4.2 Design Complexity

Provide evidence that our project is of sufficient technical complexity. Use the following metric or argue for one of our own. Justify our statements (e.g., list the components/subsystems and describe the applicable scientific, mathematical, or engineering principles)

1. The design consists of multiple components/subsystems that each utilize distinct scientific, mathematical, or engineering principles
2. The problem scope contains multiple challenging requirements that match or exceed current solutions or industry standards.

Our design requires us to have user interaction that communicates from hardware to software that is fast, responsive, and most importantly reliable. In our current design, users place physical building blocks on a conductive material, where the exact location of placement must be triangulated by using computational geometry where we must understand the electrical properties of the material so we can determine when and where a block is placed. Then, all of that information must be sent to our display, where the information is interpreted and those same buildings are visualized in a friendly and educative way that shows how that arrangement effects the efficiency and environment for the city that was built. We not only have to have quick turnarounds, use reliable interactive elements in both hardware and software that are informative, but also research and calculate the most important message we can convey with the display that is verifiable and backed by experts in urban designers and civil engineers that are on and off campus.

4.3 Modern Engineering Tools

What modern engineering tools were used for this design? Their roles.

Hardware:

3D printer – 3D prints out the required Cube (Buildings, power stations, roads).

Conductive material (touch sensitive) - As the input terminal of the entire device.

Data transfer (cable or wireless) – connects the input terminal, signal conversion system and output.

Monitor – Visualize all buildings, infrastructure, power flow, etc.

Hardware to execute our game and render visuals

Software:

SolidWorks – Build the 3D models for buildings power stations, etc.

Unity – Render the visualizations and handle logic from data being sent via the arduino

4.4 Design Context

Describe the broader context in which our design problem is situated. What communities are we designing for? What communities are affected by our design? What societal needs does our project address?

List relevant considerations related to our project in each of the following areas:

Area	Description	Examples
Public health, safety, and welfare	This project is educational, and showcases how the full-scale city can account for public health and safety.	Visualize hospitals and public places, create a more walkable city for the public
Global, cultural, and social	This project has the potential to have an impact in many regions. Since it is a planning tool to be used as a showcase, it has widespread functions.	Rural and downtown city planning, lunar colonization, planetary colonization
Environmental	This project aims to showcase a sustainability aspect. Redesigning cities can in the moment hurt the	Rooftop gardens, urban parks, solar cells, energy efficient lighting

	environment, but there are ways that the city can be built sustainably and with the environment in mind.	
Economic	This project will showcase ways to cost-effectively create a city with a model before actually setting plans for a city design or redesign.	The model itself is a low-cost system, creating these smart cities will prove cost-beneficial in the future
Stakeholders	**Description**	**Impact**
Urban Designers	Groups of people who are using this software for its intended purpose. Such as: City Planners, Park designers.	People designing cities or parks with this software will greatly benefit from the ability to iterate through designs at a higher speed. - Increased iterative design speeds.
Residents	Residents, locals, visitors, business owners, etc. Any person or group that will benefit from well-designed urban lawets.	Increasing the speed of design iterations will reduce the cost for designers to correct mistakes, and incentivize them to design urban areas that more accessible with higher utility and beauty. - Faster error correction means better designed environments

Cities	Cities, States, Countries, or any other groups that may contract the design of an environment consisting of multiple buildings.	This software will reduce the cost to design an urban style environment. - Reduced cost for contracts.

Table 2: Relevant Considerations Project Areas

4.5 Prior Work/Solutions

Skylanders / Nintendo's Amiibo is a good example of taking real life objects (such as figurines) and transferring data to software when they are placed on a playmat. These technologies use NFC to talk between the console and figure via an NFC tag on the base of the figure. With NFC, we can read and write data between a tag and a scanner. While this technology does seem similar at first to what we are doing, NFC has the inability to transfer information about where it is located on the placemat. We can still research these technologies and determine if there are elements we can take away from them that can be used in our project.

Kain, Erik. "The Rise of the Skylanders and the Big Business of Activision's Hit 'toys-to-Life' Game." *Forbes*, Forbes Magazine, 8 Aug. 2013, www.forbes.com/sites/erikkain/2013/08/07/the-rise-of-the-skylanders-and-the-big-business-of-activisions-hit-toys-to-life-game/?sh=36ee5dcd10d4.

Games like Clash of Clans are a good example of how to display information and a lawet of a city, or in this case, a village. CoC utilizes an isometric view of our settlement that lets the player see over all of the buildings in a nice cascading fashion. The buildings themselves are not 3D models, but rather pre-rendered screenshots of buildings, so rendering is not as intensive. Additionally, placing buildings is built on grids, which might be a good way to organize our rendering of the city when the user places the physical block.

Tank, Vivek. "Quickly Learn to Create Isometric Games like Clash of Clans or AOE." *Game Developer*, 7 Aug. 2018, www.gamedeveloper.com/business/quickly-learn-to-create-isometric-games-like-clash-of-clans-or-aoe.

4.6 Design Decisions

List key design decisions (at least three) that we have made or will need to make in relation to our proposed solution. These can include, but are not limited to, materials, subsystems, physical components, sensors/chips/devices, physical layout, features, etc.

1.) Physical layout of the model city will be constructed on a 1 meter by 1 meter (this size may change as the project progresses) square tile made up of scanner conductive material that is able to read the input of the figures that are placed on it and send that information to the computer monitor to visualize the city digitally. This technology is very similar to the Skylanders/Nintendo technology that the article in section 4.5 discusses. As of right now we don't quite know what specific materials will be used to make this scanner conductive material and are currently doing research to narrow this down. When this has been decide we will update this section.

2.) The models that will be placed on the 1 meter by 1 meter square tile will be made by using SolidWorks and 3-D printer. We want to tackle to art aspect of this project by constructing a model city and to design the aspects of the model city (buildings, roads, power stations) by using SolidWorks to generate layout of the specific components. Then use a 3-D printer to print out those model components. The specific types and number of those model design components has not been decided yet. We chose SolidWorks because it is a widely used CAD software that we have access to here at Iowa State and can learn to operate easily. We would then 3-D print those designs and will attach material to them that allows the 1 meter by 1 meter scanner conductive material grid to understand where everything has been placed.

3.) We will use Python as our primary coding language for the software aspect of this project. Python is a easy coding language to use that a majority of our group members know how to use. This language was primarily chosen due to personal preference over other coding languages. We will need this software to properly analyze our model city and project it onto the monitors.

4.7 Proposed Design

We have experimented with the detection of blocks using photoresistors, where we detect the absence of light which signifies that a spot is being covered up. A small demo detailing the direction of our design and project will be shown at the faculty panel presentation. This demo will show the essential aspects of our design at work. These aspects are the detection of blocks and the communication with the frontend computer that shows the current and correct state of the system, which is the current layout of the blocks. We will be showing this demo with a single grid space and various tiles.

4.7.1 Design 0 (Initial Design)

4.7.1.1 Design Visual and Description

Include a visual depiction of our current design. Different visual types may be relevant to different types of projects. We may include: a block diagram of individual components or subsystems and their interconnections, a circuit diagram, a sketch of physical components and their operation, etc

4.7.1.2 Functionality

Describe how our design is intended to operate in its user and/or real-world context. This description can be supplemented by a visual, such as a timeline, storyboard, or sketch.

How well does the current design satisfy functional and non-functional requirements?

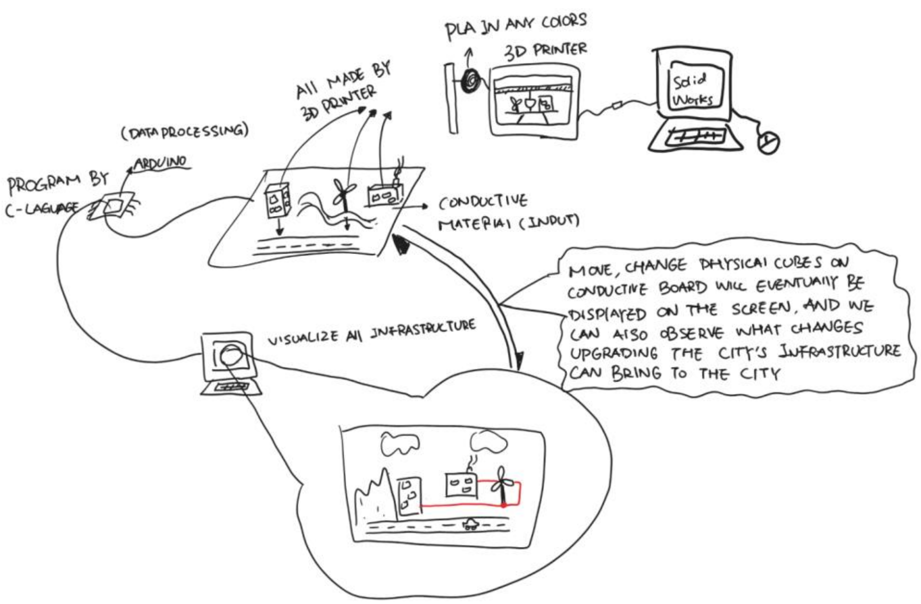

Figure 4: Interactive Urban Infrastructure Visualization Concept

4.7.2 Design 1 (Design Iteration)

Include another most matured design iteration details. Describe what led to this iteration and what are the major changes that were needed in Design 0.

The primary change to the previous design was the removal of the conductive fabric and switching to a grid based system for sensing and detecting buildings being placed. After researching how we could use conductive fabric to detect where and what kinds of buildings were being placed down, we deemed it infeasible, as we don't have much time to figure out the complicated way that could work. We created an alternative, cost effective way that would be easier to understand. This new design involves a grid with spots where you can place tiles (buildings) in. Each grid cell has 4 photo resistors on each corner of the grid cell. Each tile has a different shape and they cover a different number of photo resistors. By detecting which photoresistors are covered and which ones are not, we can derive which building is being placed, or there is no building being placed at all.

To reduce complexity of the project, we also have opted to make the buildings that are physically placed into tiles that are easy to store and manufacture. This change allows the system to be stored and transported with less effort in a form factor that a child might be more familiar with. While we are shifting our representation of the buildings from the board and focusing more on the display, we are still intending to make the tiles light up when placed on the grid. We plan to achieve this effect by utilizing the hall effect and magnets. Each tile will have a hall effect sensor paired with an inexpensive cell battery. Each grid space will have a magnet that will trigger the hall effect sensor. When the tile is placed on the grid the internal circuit will light up the LED from the power of the cell battery embedded inside the tile.

We also discovered that there were not many game engines that support Python, so we have decided to steer away from python except for extremely simple testing of basic communication protocols such as serial communication between two systems.

4.7.2.1 Design Visual and Description

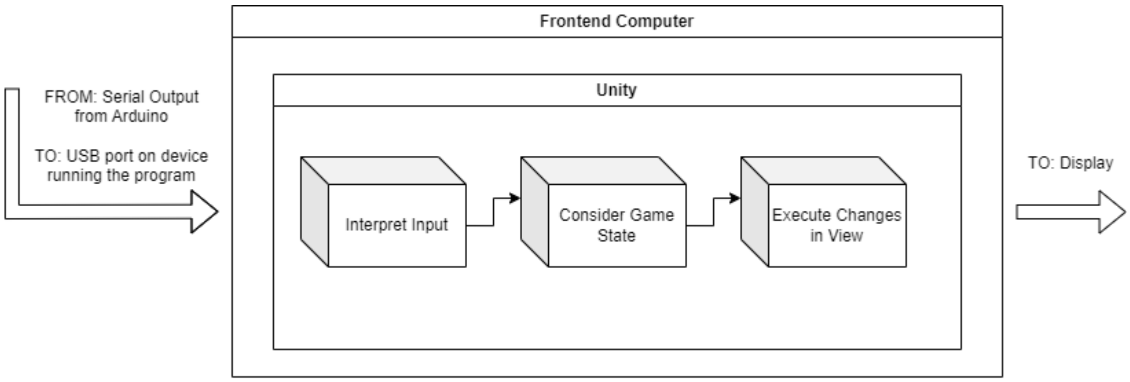

Figure 5: Arduino- Unity Interaction Flowchart

Using Unity allows us to run our logic and visualization on many different types of hardware, not just a single machine that you must maintain and keep. This download can be publicly available on GitLab or GitHub so even when we leave the university, it can be continued to be studied and used.

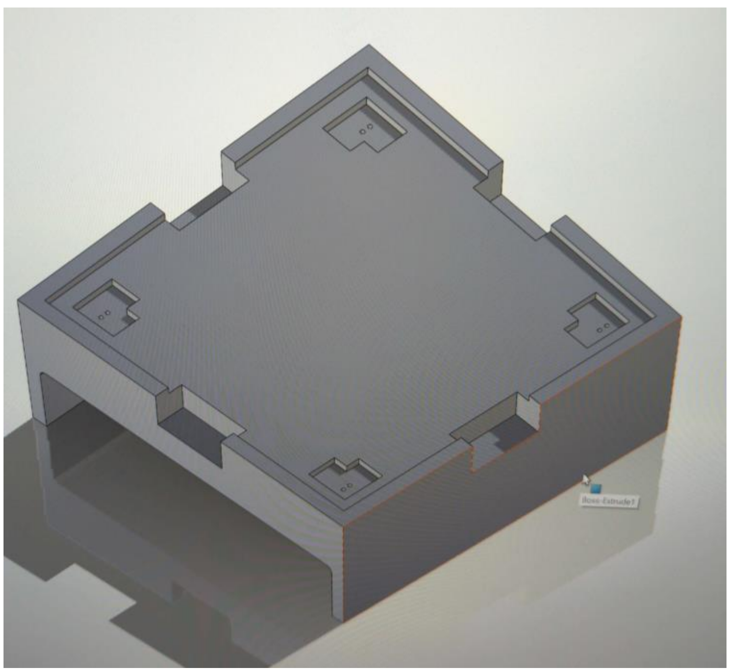

Figure 6: Single Grid Space 3D design

This is the first iteration of a single grid space. The final version of our grid will have many of these connected to each other. Each grid space is exactly 5x5 inches, with space for a 4.58 x 4.58 inch tile that has it's corners shaped according to the type of building the tile represents. The grid has 4 spaces for photoresistors that are adjusted so they aren't too embedded so enough light can hit them, preventing false detections of covering. There are holes for the pins of the photoresistor to go through and connect to a PCB which will feed to an Arduino. This is where the analog signals will be interpreted and delivered to the frontend display via a serial connection using USB.

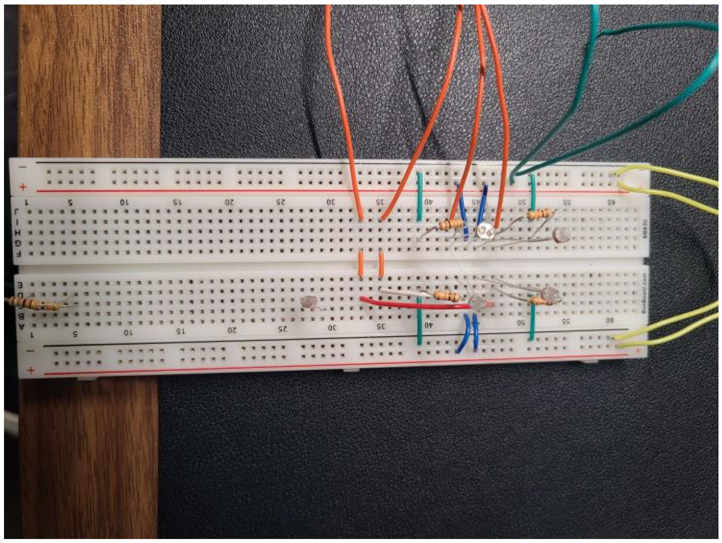

Figure 7: Unit Photoresistor model detection device

These are the internals of the demo being showcased, with the 4 photoresistors being the spots that would be covered up by a tile.

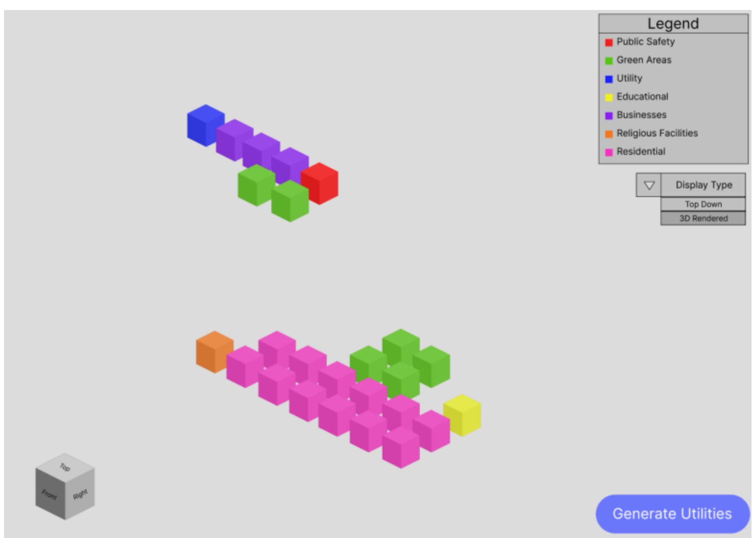

Figure 8: Rendered Output on The Display

This is an example of the rendered output on the display. Right now, each building is represented by a single-colored block, but we want to have each building look like how they look in real life. You can see the task that is prompted to the user in the bottom right corner. This gives a goal and incentive to keep playing around with the architecture and see how well you can build a great town.

4.8 Technology Considerations

We chose to use Unity for the frontend visualization of buildings being placed down and game logic. The reasons Unity was selected can be divided into two categories – function and non-functional reasons.

Functionally, our project requires a real time method of representing objects in a 3d environment that reflects user input (I.e. block placements on the playmat). These specific requirements steered us towards using a game engine as developing these tools from the ground up on a different platform would not only be difficult and time consuming for us to develop, but it would be nearly impossible to maintain. Game engines have a large number of internal tools with exhaustive documentation that lets anyone that is familiar with the engine platform unpack and understand the game logic and perform changes. Additionally, game engines allow the easy export of projects so they can be executed on a wide variety of different machines with different architectures. This feature allows our project to be more portable and not require a specific computer or operating system to run the game. For example, it can be run on a Linux desktop attached to a monitor just as well as a Macbook or a Windows tablet. As long as the hardware has a display and the correct I/O (in this instance a USB port) you can connect the playmat and have the system functional.

Unity was also selected for non-functional reasons, such as cost and learning curve associated with developing with the platform. Unity is free to download and develop for, saving us from draining our $500 budget for senior design. Our software engineers Ally and Grant will be the two software engineers working on the visualization front end the most, so it is important to consider their skillsets when selecting a game engine. Ally and Grant are both familiar with C#, the same language that Unity uses for scripting logic and networking. Having this familiarity will make development on the Unity platform faster for both of our software engineers.

To create our 3d models that are to be displayed in an isometric angle in Unity, we chose to go with an open-source project called MagicaVoxel; which is an intuitive tool for creating voxel-based models. Voxels can be described as the three-dimensional version of a pixel, where each voxel block is exactly 1 x 1 x 1. You can recognize voxels in games such as Crossy Road and Minecraft. Utilizing voxels instead of modeling highly detailed buildings has many advantages. For one, models made out of voxels are very easy to edit and export, which lowers the barrier of entry for everyone on our team to make their own building designs and contribute to the creative process. Secondly, they are easy to render, which allows the program to run well on lower-performance, less expensive hardware. Voxels also offer a friendly and endearing aesthetic that is friendly and understandable for a wide audience that includes children.

4.9 Design Analysis

One aspect of our design that did not work initially was the model of the grid space. As it was designed, it was difficult to 3d print because it requires a lot of supports to print which increases the waste created from manufacturing and turnaround time for printing. A new grid space will be designed to make the connection of tiles more seamless and elegant while making the job of our electrical engineers easier. This is an aspect of our project that we need to keep track of throughout the next semester. If we make too many compromises for the sake of ease, our installation will not look as good as it could.

5 Testing

Testing is an extremely important component of most projects, whether it involves a circuit, a process, power system, or software.

Units being tested include the conductive material's electrical properties, computational geometry algorithms, and the digital display algorithms. Tools might include electrical testing tools and software debugging tools.

Given the complexity of both hardware and software components, the testing strategy will focus on individual unit tests, interface tests between components, integration tests, system-level tests, regression tests, and acceptance tests. A unique challenge in testing is ensuring real-time responsiveness and accuracy in detecting block placements and translating them to the digital display.

The testing plan should connect the requirements and the design to the adopting test strategy and instruments. In this overarching introduction, given an overview of the testing strategy. Emphasize any unique challenges to testing for our system/design.

The primary interfaces in the design are between the block placement detection system and the digital display. Testing will ensure that data is correctly and swiftly transmitted from the hardware to the software. Tools would include data logging and visualization tools.

5.1 Unit Testing

What units are being tested? How? Tools?

Units being tested:

- **Hardware:**
  - **Surface**
    - Durability Tests: These tests assess the surface's resistance to scratches, dents, and bends. Methods might include hardness testing, drop tests, and bend tests.
    - Environmental Tests: To check the surface's resilience to various environmental factors like humidity, temperature extremes, and UV exposure, environmental chambers are used.
    - Aesthetic and Ergonomic Assessment: This involves evaluating the surface's appearance, texture, and how comfortable it is to handle. User studies might be conducted for this purpose.
  - **Processor**

        - Performance Testing: Benchmarking the processor using various tools and software to assess its speed, efficiency, and ability to handle multiple tasks simultaneously (multi-threading).
        - Stress Testing: Running the processor at full capacity for extended periods to ensure stability and to check for overheating issues.
        - Power Consumption and Heat Dissipation: Testing how much power the processor consumes under various loads and how effectively the heat is dissipated.
    - **LED/ LCD/ OLED Display**
        - Color Accuracy and Brightness Tests: Using colorimeters and light meters to assess color accuracy, brightness levels, and contrast ratios.
        - Viewing Angle Tests: Checking how the display's visibility and color consistency change at different angles.
        - Touchscreen Functionality: If the display is touch-sensitive, tests are conducted to assess the responsiveness, accuracy, and multi-touch capabilities.
    - **Input/output ports**
        - Connectivity Tests: Checking each port's ability to establish and maintain a connection with various peripherals.
        - Data Transfer Speed Tests: Measuring the speed and reliability of data transfer through the ports.
        - Durability and Wear Tests: Repeatedly plugging in and unplugging devices to ensure the ports can withstand regular use.
        - Electrical Testing: Ensuring that the ports provide correct power output and data signals.

- **Software:**
    - **Analog to digital conversion (De-modulization)**
        - Analog to Digital Conversion (ADC) is the process of converting continuous analog signals from our touch surface into discrete digital values. This is a critical step in digital signal processing where real-world analog user inputs are converted into a digital format that can be processed by our software.

- Accuracy Testing: Compare the digital output with the expected values for given analog inputs to ensure accurate conversion. This can involve using signal generators and oscilloscopes.
- Noise and Interference Testing: Introduce noise or interference in the analog signal to verify how well the ADC handles real-world imperfect signals.
- Range and Resolution Testing: Test the ADC across its entire input range to ensure consistent performance.

- **Data input**
  - Data input refers to the process of receiving data from external sources for further processing. This will happen via user input through sensors in our touch surface and through mouse and keyboard input devices.
  - Input Validation Testing: Ensure that the software correctly accepts and processes valid inputs from touch surfaces, mice, and keyboards, and rejects invalid inputs.
  - Stress Testing: Simulate high-frequency inputs or large volumes of data to test the system's ability to handle heavy loads.
- **Data Scrubbing**
  - Data scrubbing, also known as data cleaning, will involve identifying and correcting (or removing) errors and inconsistencies from data to improve its quality. This step is crucial in ensuring that the subsequent data analysis is accurate and reliable.
  - Data Quality Assessment: Introduce datasets with known errors or inconsistencies and verify that the scrubbing process identifies and corrects these issues.
  - Performance Testing: Measure the efficiency of the data scrubbing process, especially when handling large datasets.
- **Data Parsing:**
  - Data parsing is the process of taking input data and converting it into a more usable format. This will involve interpreting and separating the data stream into discrete data packages that can be used in the processing stage according to a set of predefined rules or formats.
  - Functional Testing: Test the parser with a variety of input data formats to ensure it correctly interprets and separates data as expected.

        - Error Handling: Verify how the parser reacts to malformed or unexpected data, ensuring it fails gracefully or throws appropriate exceptions.
    - **Data Processing**
        - Data processing will the transformation of parsed data into meaningful information through a series of actions or operations. This will include data classification based on input source. It will also sort and tag data based block type and location on the touch surface.
    - **Data Modeling**
        - Data modeling will be the process of taking processed data and translating it into the in memory model of the surface. This will be emplacing the blocks onto the virtualized surface.
        - Model Verification: Compare the in-memory model generated by the software with the expected model for given inputs.
        - Performance and Scalability Testing: Assess how the data modeling process scales with increasingly complex or large datasets.
    - **Rendering**
        - Rendering will be the process of generating a final image from the modeled data. This term will be graphics and visualization, where data is converted into a graphical representation.
        - Visual Inspection: Manually inspect the rendered images to ensure they visually match the expected output.
        - Automated Image Comparison: Use software tools to compare rendered images with a set of reference images for consistency.
        - Performance Testing: Evaluate rendering times and resource usage, particularly under different hardware configurations.

5.2 Interface Testing

What are the interfaces in our design? Discuss how the composition of two or more units (interfaces) are being tested. Tools?

The interfaces in our design are the block grid, the blocks themselves, the processing unit, and the display. The block grid will be tested manually and with extreme values to ensure that the blocks are being sensed correctly. The display will be tested with automatic software testing that simulates blocks being placed to ensure that the correct block is being displayed in the correct place. It will also ensure that no buildings can be placed on top of each other.

To ensure new additions or modifications don't break existing functionalities, automated testing suites will be employed after manual tests are verified. Critical features to monitor include the accuracy of block placement detection and the correctness of the educational display. These tests will be driven by design requirements.

Key Hardware Interfaces in the Design:

- **Testing Surface Hardware Interfaces:**
  - Connector Durability Testing: If the surface includes connectors (like a detachable keyboard or accessories), test the durability and reliability of these connectors.
  - Signal Integrity Testing: For surfaces that interact with other components (like touchscreens), test the quality and reliability of the signal transmission between the surface and the control unit.

- **Testing Processor Hardware Interfaces:**
  - Socket/Connection Testing: Test the physical connection between the processor and the motherboard (like the CPU socket) for proper contact and signal transmission.
  - Bus Interface Testing: Evaluate the communication interfaces (like PCIe, USB, etc.) to ensure they transmit data correctly and at the expected speeds.

- **Testing Display Hardware Interfaces:**
  - Display Connector Testing: Test the connectors (like HDMI, DisplayPort, VGA) for signal transmission quality and physical durability.
  - Touch Interface Testing: For touchscreens, test the interface that connects the touch panel to the processing unit, ensuring accurate and consistent touch data transmission.

- **Testing Input/Output Port Hardware Interfaces:**
  - Port-to-Motherboard Testing: Check the connection between the I/O ports and the motherboard or the main control unit, ensuring stable and correct data transfer.
  - Peripheral Compatibility Testing: Test compatibility with a range of peripheral devices to ensure the ports can communicate with a wide array of external hardware.

- **General Hardware Interface Testing Methods:**
    - Electromagnetic Compatibility (EMC) Testing: Ensure that electromagnetic interference does not affect the performance of the interfaces.
    - Stress Testing: Subject the interfaces to high data rates, continuous operation, and extreme conditions to assess their durability and performance under stress.
    - Environmental Testing: Expose the interfaces to various environmental conditions like temperature extremes, humidity, and vibration to ensure they operate reliably under different scenarios.

- **Key Software Interfaces in the Design:**
    - Analog-to-Digital Conversion Interface:
        - The interface between the physical touch surface (hardware) and the ADC software component.
    - Data Input Interface:
        - The interface through which user inputs (from touch surface, mouse, keyboard) are received by the software.
    - Data Scrubbing Interface:
        - The interface between the data input stage and the data scrubbing/cleaning module.
    - Data Parsing Interface:
        - The interface between the data scrubbing module and the data parsing module.
    - Data Processing Interface:
        - The interface between the data parsing module and the data processing logic.
    - Data Modeling Interface:
        - The interface between the data processing module and the data modeling component.
    - Rendering Interface:
        - The interface between the data model and the rendering engine.

- **Testing Composition of Interfaces:**

- When testing the composition of two or more interfaces, the focus is on integration testing. This involves verifying that data flows correctly between different modules and that these modules interact as expected.

- **Scenario-Based Testing:**
  - Create real-world scenarios to simulate how different components interact.

- **Tools for Testing:**
  - Automated Testing Frameworks:
    - Tools like Selenium, Postman, or JUnit can be used for automating interface tests.
  - Mocking and Stubbing Tools:
    - Tools like Mockito or WireMock can simulate parts of the system (like external services or data sources) that are not being tested.
  - Performance Testing Tools:
    - Tools like JMeter or LoadRunner to test how interfaces perform under load.
  - Static Code Analysis Tools:
    - Tools like SonarQube can be used to analyze the integration points for potential issues in code quality or security vulnerabilities.
  - Continuous Integration Tools:
    - Platforms like Jenkins or Travis CI to automate testing as part of the development pipeline.

- **Testing Data Flow and Dependencies:**
  - Ensure that data passed between interfaces is correct both in format and content.
  - Test how changes in one module affect the downstream modules (e.g., how data scrubbing impacts data parsing).

- **Testing Error Handling and Failures:**
  - Simulate failures at various interfaces to test the system's robustness and error handling capabilities.
  - User Acceptance Testing (UAT):

- Review implementation and test with target audience (younger children) to ensure that our goals are being met and our messaging is coming across in a fun and joyful manner.

5.3 Integration Testing

What are the critical integration paths in our design? Justification for criticality may come from our requirements. How will they be tested? Tools?

To demonstrate that design requirements are met, end-to-end tests will be conducted with representative users placing blocks and receiving feedback from the display. Client involvement would include regular demonstrations and feedback sessions to ensure the project aligns with their expectations.

Critical integration paths include the real-time transmission of block placements to the digital visualization and the integration of research data into the educational display. These paths will be tested for speed, accuracy, and reliability using software test suites and user testing.

5.4 System Testing

Describe system level testing strategy. What set of unit tests, interface tests, and integration tests suffice for system level testing? This should be closely tied to the requirements. Tools?

Strategy:

For system-level testing, a combination of previously performed unit, interface, and integration tests will be employed. The goal is to ensure that when all components work together, the system behaves as expected. The emphasis will be on real-time responsiveness, accuracy in block placement detection, and the correctness of the educational visualization.

Tools:

- Real-time Monitoring Tools: To assess system response times.
- Data Visualization Tools: To ensure data flow from block detection to display is correct.
- User Feedback Tools: To get real-time feedback from users during system tests.

5.5 Regression Testing

How are we ensuring that any new additions do not break the old functionality? What implemented critical features do we need to ensure do not break? Is it driven by requirements? Tools?

To ensure new additions or modifications don't compromise existing functionalities, an automated testing suite will be set up. This suite will run previous tests on the system after every significant change.

Critical Features:

- Accuracy of block placement detection.
- Correctness of the educational visualization.
- Responsiveness of the system.

Tools:

- Automated Testing Frameworks: To automate the regression testing process.
- Version Control Systems: To track changes and easily roll back if issues arise.

5.6 Acceptance Testing

How will we demonstrate that the design requirements, both functional and non-functional are being met? How would we involve our client in the acceptance testing?

To demonstrate that design requirements are met, end-to-end tests simulating real-world scenarios will be conducted. Users will place blocks, and the system's response and educational feedback will be assessed.

The client will be actively involved in this phase, participating in demonstrations and providing feedback. Their insights will ensure that the system not only meets technical requirements but also fulfills the intended educational purpose.

For non-functional requirements, we evaluate the system's performance under different conditions, perform security tests to check if system is protected against unauthorized access, and other security threats.

5.7 Security Testing (if applicable)

Given that this is an interactive installation, direct security threats may be minimal. However, to ensure the safety of the system and its users:

- The software will be tested for vulnerabilities, especially if there's any data collection or network connectivity.
- The hardware components will be assessed for safety standards to prevent electrical hazards.

5.8 Results

Results from the various testing phases will be thoroughly documented, with emphasis on how they ensure compliance with the project requirements. Visual aids like graphs, tables, and flow diagrams will illustrate the testing process and outcomes. The narrative will conclude with an assessment of the design's alignment with its intended purpose and requirements.

6 Implementation

- **Photoresistor Grid**
    - Acquire all necessary hardware
    - Assemble all modules
    - Ability to send all data to the Arduino correctly
- **Buildings**
    - 3D print all of the buildings
    - Add LED lights to light up when placed onto the grid
- **Arduino**
    - Acquire the Arduino and all necessary shields
    - Program the Arduino to process and send all data
- **Frontend device**
    - Acquire device
    - Ensure device has necessary updates to run everything
- **Unity Application**
    - Frontend
        - Create an interface that will display data correctly
        - Display data dynamically

- Backend
    - Ability to connect to the Arduino
    - Ability to process data from the Arduino

7 Professionalism

This discussion is with respect to the paper titled "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", International Journal of Engineering Education Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

Area of Responsibility	NSPE Definition	Corresponding Code of Ethics
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence	IEEE: As engineers we have to maintain high standards in all areas of our own work. The IEEE and NSPE definitions are very similar the only difference is that IEEE also includes ethical conduct in addition to professional conduct. I. To uphold the highest standards of integrity, responsible behavior, and ethical conduct in professional activities.
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	IEEE: As engineers we must follow national and international laws and regulations, and not take bribes, and not to cheat customers or take advantage of them. The IEEE and NSPE definitions are very similar the only difference is that NSPE emphasizes the deliverance of products and services at a reasonable cost. I4. To avoid unlawful conduct in professional activities, and to reject bribery in all its forms.
Communication Honesty	Report work truthfully, without deception, and are understandable to stakeholders.	IEEE: We as engineers must be able to give and receive fair criticism, correct any issues that arise, be truthful about data and information, and give credit where credit is due. The IEEE and NSPE definitions are very similar

		however IEEE goes more into detail about the manner in which engineers need to hold one another accountable. I5. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, to be honest and realistic in stating claims or estimates based on available data, and to credit properly the contributions of others;
Health, Safety, and Well-Being	Minimize risks to safety, health, and well-being of stakeholders.	IEEE: We as engineers need to respect safety, privacy, well-being, and the environment. The IEEE and NSPE definitions are very similar however IEEE includes more that just stakeholders as it addresses people and environment in general. I1. to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment;
Property Ownership	Respect property, ideas, and information of clients and others.	IEEE: We as engineers cannot do anything that will cause physical, mental, or reputational harm to others. The IEEE and NSPE definitions are very similar but the difference is that IEEE also includes the reputation of others. I9. to avoid injuring others, their property, reputation, or employment by false or malicious actions, rumors or any other verbal or physical abuses;

Sustainability	Protect the environment and natural resources locally and globally.	IEEE: We as engineers need to have empathy for the public. We must think of ethical design processes. We must disclose information that users or consumers should know. The IEEE and NSPE standards are very similar however one difference is that NSPE focuses primarily on the environment and IEEE focuses on general safety, health and welfare of people and environment. I1. to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, to protect the privacy of others, and to disclose promptly factors that might endanger the public or the environment;
Social Responsibility	Produce products and services that benefit society and communities.	IEEE: We as engineers need to help society in general understand the technologies we are creating. The IEEE and NSPE definitions are very similar however IEEE focuses on not only producing products that benefit society but also educating society on said technologies. I2. to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems;

Table 3: Areas of Responsibility

7.2 Project Specific Professional Responsibility Areas

For each of the professional responsibility area in Table 1, discuss whether it applies in your project's professional context. Why yes or why not? How well is your team performing (High, Medium, Low, N/A) in each of the seven areas of professional responsibility, again in the context of your project. Justify.

Area of Responsibility	NSPE Definition	Project Application
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence	High – This is our first major project as students so we are following all the procedures and guidelines to implement our project. We are constantly getting are project checked by faculty members to ensure work quality is being met and the highest standard of professionalism is being upheld. It is important for us to have integrity to build our reputation as engineers in the workplace.
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	Low – We are not really making a sellable product, nor are we working with a large budget so everything will be within a reasonable cost. Our project is also more research based than us implementing a sellable product. We still need to be mindful and reasonable with the cost of items for our project.
Communication Honesty	Report work truthfully, without deception, and are understandable to stakeholders.	High – We are working in a team on this project so our communication skills are paramount to our success in this project. We must be honest about our work with one another and also be honest with our faculty advisor, and the general public. We must not create deception in our work to pass it off as more interesting to the general public or more/less complex to our advisors. We must be truthful in communicating our goals and issues with all audiences.

Health, Safety, and Well-Being	Minimize risks to safety, health, and well-being of stakeholders.	Medium – Health and safety is always important. However our project isn't really dangerous or harmful to stakeholders. We will create and implement our project in a smart and safe manner that ensures the safety of not only our team members but also the general public. When we make the physical model of our city we will make sure that all safety standards are being met.
Property Ownership	Respect property, ideas, and information of clients and others.	High– We will respect the ideas of our fellow teammates and faculty members. We will respect the input from the general public on our project. We will also give credit where credit is due throughout the implementation of our project by siting the sources we used to get inspiration.
Sustainability	Protect the environment and natural resources locally and globally.	Medium – Our project encompasses sustainability of smart cities by finding ways to improve urban infrastructure. Our research is mainly about this however when we actually make our model we don't really have to worry about it being sustainable because of it's small size. Our model city will not cause much harm on the environment.
Social Responsibility	Produce products and services that benefit society and communities.	High – The purpose of this project is to produce a model city that can be displayed publicly to inform the general public about ways to improve urban infrastructure. We want to produce a high quality product that can convey this to the general public.

Table4: Specific Professional Responsibility Areas

7.3 MOST APPLICABLE PROFESSIONAL RESPONSIBILITY AREA

The most applicable Area of Responsibility for our project is Social Responsibility. The entire purpose of our project is to create an interactive public display that educates the public about ways to improve urban infrastructure. Our interactive model city will allow us to convey this information to the general public.

8 Closing Material

8.1 DISCUSSION

Discuss the main results of your project – for a product discuss if the requirements are met, for experiments oriented project – what are the results of the experiment, if you were validating a hypothesis – did it work?

Our project, Engineering Art, asked us to select a problem from 14 Modern Problems that Engineers Currently Face and to implement it in an artistic manner the utilizes and showcases all of our engineering majors. We selected Ways to Improve Urban Infrastructure, and decided to represent it in an artistic manner by creating an interactive model city that the general public can interact with and gain knowledge on how urban infrastructure can be improved. During this project we made use of the double diamond process. Our project is different from other groups because initially we were just given a general direction rather than a defined project with a specific set of constraints. We had to discover and define what we wanted our project to be before we got to solving the problem. Due to this, a majority of our time this semester was spent conducting primary and secondary research on our issue and then define the project we wanted to do. After we defined the project we began developing it. This is where we are currently at in our project timeline and this is where we expected ourselves to be at this time. As of right now all of our initial project requirements are being met and we are on schedule. The requirements of the overall Engineering Art project is being met as we have selected a problem and are representing it in an interactive artistic manner that utilizes and showcases all of our engineering majors. We are also meeting all of the project requirements that we defined for ourselves in the sections above in this design document.

8.2 CONCLUSION

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals. What constrained you from

achieving these goals (if something did)? What could be done differently in a future design/implementation iteration to achieve these goals?

Due to the uniqueness of our project, much of the work we have done this semester was essentially to “catch up” to the other groups. We were given a very vague set of instructions to look over 14 Modern Problems that Engineers Currently Face and choose a problem to implement in an artistic manner the utilizes and showcases all of our engineering majors. We selected Ways to Improve Urban Infrastructure, and decided to represent it in an artistic manner by creating an interactive model city that the general public can interact with and gain knowledge on how urban infrastructure can be improved. We conducted a lot of primary and secondary research on our topic before defining our project and set of constraints. We are currently in the phase of development, as we are working towards the development of our model city. Our goals for this project were as follows. Read through the 14 Modern Problems that Engineers Currently Face and choose one problem to solve. Conduct research on said problem and come up with a solution. Define how the problem can be implemented in an artistic manner than makes use of all our engineering majors. Then work towards developing a prototype. Currently all of our goals have been met or are in the process of being met, referring to the ongoing development of our prototype model. The main thing that constrained us was the lack of constraints. Due to our project being so open ended, we had do conduct a lot of research on a topic before we could even define a set of constraints for what we wanted our project to be. This was very confusing to do because we didn’t quite know how to define a doable project that tackles our issue while also meeting the requirements of the Engineering Art project guidelines. However after frequent meetings with our advisor/client we were able to hash these issues out and actually begin working on implementing our actual project. This process set us back by quite some time as the research and define portion of our project took a long time. If we could do this project again we would try to be more on top of things during the beginning of the semester so that we could have selected our project faster and defined it’s constraints so that we could being implementing it sooner.

8.3 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

Luna Optoelectronics, “CdS Photoconductive Photocells,” PDV-P9200 datasheet, Oct. 2016

8.4 Appendices

Any additional information that would be helpful to the evaluation of your design document. If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.

8.4.1 Team Contract

Team Name sdmay24-17

Team Members:

1) Karandeep Sandhu

2) Alexandra Whitmer

3) Grant DeWaay

4) Zheyuan Zhang

5) Austin Krekula

6) Seyedehbahareh Hashemimovahed (Bahar)

Team Procedures

3. **Day, time, and location (face-to-face or virtual) for regular team meetings:**

   Thursday from 10 a.m. to 11 a.m. in the Student Innovation Center.

4. **Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):**

Discord and Cisco Webex and we also have everyone's phone numbers in case we need to text one another.

5. **Decision-making policy (e.g., consensus, majority vote):**

We will do a majority vote as we want to make sure that everyone's voice is heard and we take everything into consideration.

6. **Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):**

We will all be responsible for tracking our own individual progress and how much time we are spending on the project. As for team meetings we have someone in our group (yet to be determined) take note of what was discussed during the meetings. We will also be using GitLab and google docs and Cybox to store our documents and project material.

Participation Expectations

7. **Expected individual attendance, punctuality, and participation at all team meetings:**

a. We expect everyone to attend the all the meetings, however we have noticed that not everyone can meet during our work week meeting time due to conflicting schedules so we will schedule meetings on the weekend if need be. However we will keep group members who cannot make to the meetings in the loop by constantly communicating via Discord and Webex. Obviously if there are any conflicts that arise for group members we can adapt and keep them in the loop via discord and webex. We expect everyone to be respectful to one another and participate equally in this project process. we should not have a few group members do all of the work while others do very little and slack off. We expect everyone to do what was assigned to them in a timely manner however if issues come up we as a group will adapt.

8. **Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:**

b. We expect everyone to do what was assigned to them in a timely manner however if issues come up we as a group will adapt. We plan on hitting deadlines regularly so we don't fall behind.

9. **Expected level of communication with other team members:**

c. We make use of discord and webex and expect everyone to check the discord at least once a day to ensure nothing unplanned has popped up and everything is moving as scheduled. If something unexpected pops up and needs to be handled immediately then we will communicate via text or groupme as those might generate faster responses.

10. **Expected level of commitment to team decisions and tasks:**

d. As mentioned above we will use a majority vote to decide what needs to be done and expect everyone to participate in generating ideas, providing feedback and voting. We have a team advisor who will help guide us so they might tell us the direction we need to go but all team decisions will be made by the members.

Leadership

11. **Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):**

e. Everyone will participate in our team and we all expect that as tasks arise we will distribute them to meet everyone's specific skill set and availability. We have listed everyone's skill sets below. Our client is our advisor so we will meet with them on weekly basis.

f. Artistic and creative direction that includes non technical elements will be primarily handled by Grant DeWaay in addition to anyone else on the team that expresses interest in contributing to this aspect of the project. This role may transfer to anyone else in the group if there is overwhelming distaste of the ideas being presented, or the practices being implemented. Elements that fall under this category include the following:

i. Shape and form

ii. Color palette

iii. Materials used for non technical elements
12. ex: the enclosure
iv. Presentation
v. Messaging
13. NOT including the selection of 1 of the 14 engineering challenges
14. NOT including changing project scope for the sake of greater artistic expression unless agreed upon by the entire group

15. Strategies for supporting and guiding the work of all team members:

g. As mentioned before we have team members from all four of the engineering fields (computer, electrical, software, cyber) and we will use our individual skill sets to help each other succeed. We will try to divide the work fairly and also in a way that allows everyone to work with something they have experience in. If issues arise then we will help each other figure them out.

16. Strategies for recognizing the contributions of all team members:

h. During our weekly meetings we will all touch base and share our individual progress and will be able to see how much work someone is doing by tracking their progress on their tasks.

Collaboration and Inclusion

17. Describe the skills, expertise, and unique perspectives each team member brings to the team.

Zheyuan: Matlab, AutoCAD and some simple C language.

Alexandra: Java, frontend development, and some hardware experience.

Karandeep: Circuit design, Matlab, Cadence Virtuoso, and some Verilog and simple C.

Grant DeWaay: Wide array of programming languages along with modeling tools like Solidworks and Grasshopper 3D and manufacturing techniques like 3D printing, sewing, embroidery, and laser cutting. Experience with double diamond design model and design thinking.

Austin Krekula: Security oriented design thinking, information security, risk management framework, Linux OS security, Windows OS security, networking security, applications of cryptography, discrete mathematics, Python, C, Java, Verilog, assembly language (MIPS & x87 instruction sets), software reverse engineering, digital forensics, embedded systems engineering, microprocessor design, digital logic, algorithm analysis.

Bahar: Java, some simple C, some embedded system and VHDL.

18. Strategies for encouraging and support contributions and ideas from all team members:

i. Establish an open communication environment, encouraging team members to share their thoughts and opinions.

j. Ensure everyone has an opportunity to speak.
k. Strengthen trust and collaboration among teams through long-term cooperation.
l. Ensure everyone understands their tasks and expectations.

19. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)

Our team discord is a space for any thoughts and ideas. Please feel free to share if something is making you uncomfortable or unable to complete a task. These messages should be taken without criticism, and we can redirect our team from there.

Goal-Setting, Planning, and Execution

20. Team goals for this semester:

Have our project planned out and reach the step where we begin ordering parts. Have a good design document made and do well on the presentation to receive a good grade (A) for the semester.

21. Strategies for planning and assigning individual and team work:

m. We want to distribute work in a fair manner so that everyone does a relatively equal amount of work and we also want to assign that work based on everyone's strengths as mentioned above.

22. Strategies for keeping on task:

n. We will use our weekly meeting time to go over progress and make sure everyone is contributing and staying on task. For individual strategies that will vary from person to person so encourage all team members to work in a way that works for them without jeopardizing the team.

Consequences for Not Adhering to Team Contract

23. How will you handle infractions of any of the obligations of this team contract?

We will reach out to the person to check in. If the person is unresponsive, we will divide the task among other team members and consult with our faculty advisor or TA if need be.

24. What will your team do if the infractions continue?

We will set a time for a group team meeting where we will discuss group conduct and resolve issues. If that does not work we will inform our TA or instructor and go from there.

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*

b) *I understand that I am obligated to abide by these terms and conditions.*

c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) Karandeep Sandhu DATE 9/10/2023

2) Alexandra Whitmer DATE 9/10/2023

3) Grant DeWaay DATE 9/10/2023

4) Zheyuan Zhang DATE 9/10/2023

5) Austin Krekula DATE 9/10/2023

6) Seyedehbahareh Hashemimovahed DATE 9/10/2023