IOWA STATE UNIVERSITY

CyWi: An Open-Source Wireless Innovation Lab for SmartAg, AR/VR, and Beyond

Project Plan Document

Chenye Lim *Ryan Cullinan*

Jian Chew *Shay Willems*

Pawel Darowski *Tyler Beder*

SD-DEC19-02

April 2019

I. Frontal Material

Table of Contents

List of Figures

List of Tables

II. Introductory Material

Problem Statement

Over the last decade, Internet of Things (IoT) technologies have been extensively studied due to the many advantages that interconnectivity can bring to everyday life. Despite many years of research, we have barely begun witnessing the capabilities of Industry 4.0 and IoT. These fields have not been penetrating our worlds as fast as previously predicted. IoT has the potential to be useful for a wide range of applications such as SmartAg, connected autonomous vehicles, smart grids, and augmented/virtual reality (AR/VR). Powered with the capabilities of 5G mobile, IoT promises to change every aspect of our lives. But to expand the pace of innovation, researchers need access to state-of-the-art spaces to run experiments and gather real-world data.

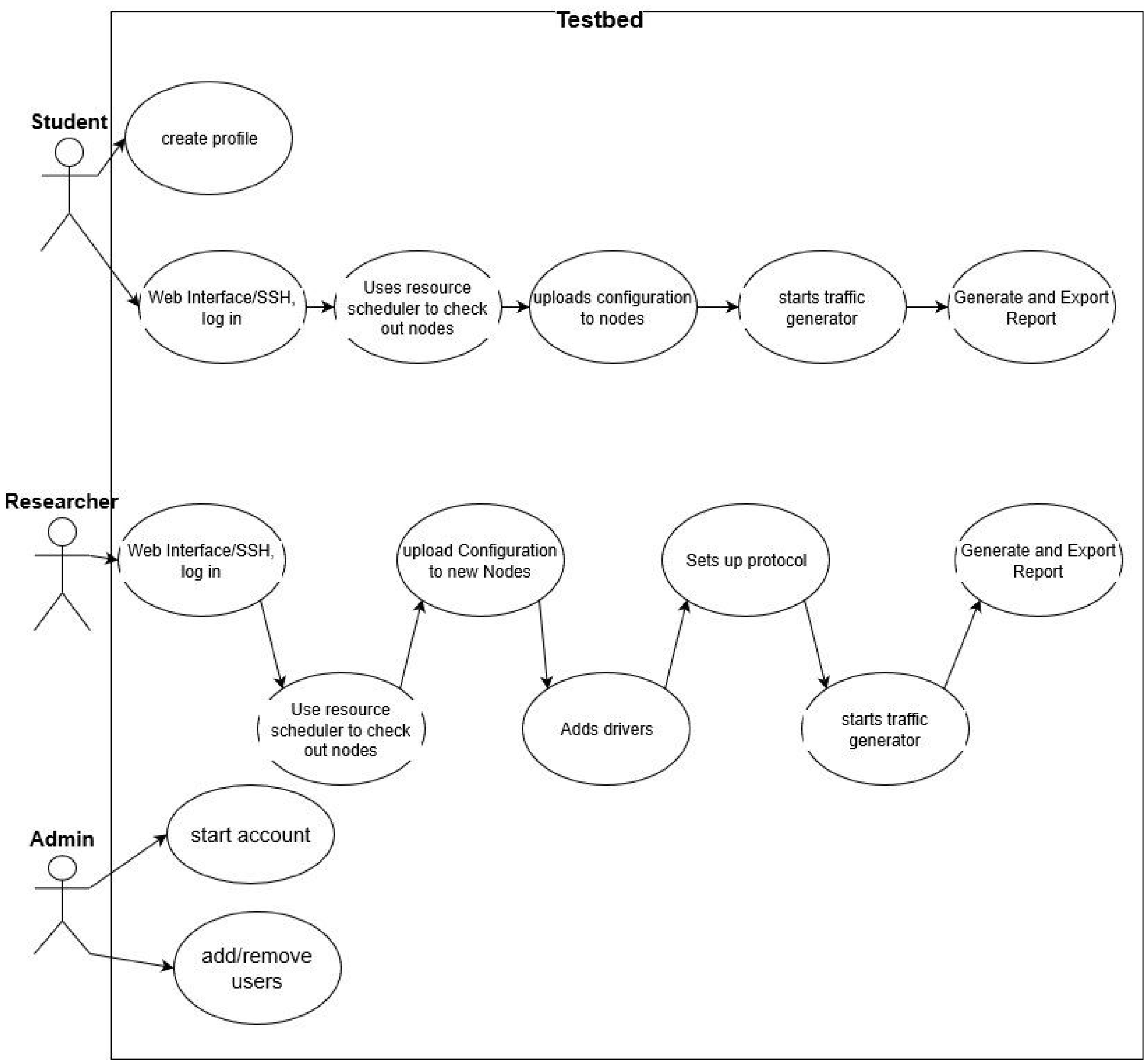

Figure 1: Use Case Diagrams

Iowa State University has decided to explore this field by developing the CyWi lab. This lab will consist of a multi-node wireless testbed located on the Iowa State campus. As seen above in Figure 1, the testbed will have many use cases for students and researchers alike. Users will be able to freely run their custom experiments via remote access. Experiments can range widely but will include performance analysis, cyber security demonstrations, and protocol testing. CyWi will feature OpenAirInterface, an innovation platform for the standardization of 5G mobile. Popular IoT protocols such as Bluetooth Low Energy and Zigbee will also be available for research. Our hope is that providing a free and open-source research environment to the world will stimulate faster innovation in wireless communications.

Operating Environment

The CyWi's testbed will be located in 3050 Coover Hall at Iowa State University, a climate-controlled room with a keycard-secured door. The only people allowed into the lab room will be the professor/client, the Electronics Technology Group (ETG), and a handful of researchers and developers. Users will only have access via the remote web interface, never via physical access to the lab. One wall has windows facing west but the blinds will be closed so sunshine never touches our equipment. While it is not expected for the hardware to experience anything but optimal operating conditions, our web-based service will be exposed to the Internet so cyber attacks and DDoS are possible.

Intended Users and Intended Uses

The CyWi testbed is intended for two general types of users: students and researchers.

Students learn about wireless signals and protocols in their courses via lectures, assignments, and small projects. Theoretical knowledge is crucial. However, building upon that foundation with extensive lab experience configuring real-world hardware will improve students' understanding of the subject matter. Implementing wireless topologies such as mesh, star, and point-to-point could inspire future IoT developers. Comparing Zigbee and Bluetooth Low Energy performance, for example, over assorted ranges, signal strengths, and conditions will extend students' knowledge of technology strengths and limitations. CyWi will provide students the opportunity to experiment with a variety of popular, existing wireless technologies and to expand their understanding in a safe environment.

Researchers, on the other hand, will appreciate the cutting-edge communication technologies represented by the CyWi testbed. Access to powerful and configurable software-defined radios (SDRs) will allow researchers to study a wide spectrum of new heterogeneous networks and explore exciting innovative ideas. Researchers will evaluate performance monitoring and statistics after each experiment to determine feasibility of their chosen path. Emerging technologies such as 5G, SmartAg, augmented reality, virtual reality, and Internet-of-Things will generate opportunities for decades of continuous communications development. As Figure 2 shows, CyWi is particularly equipped to handle experiments of various throughput and latency specifications.

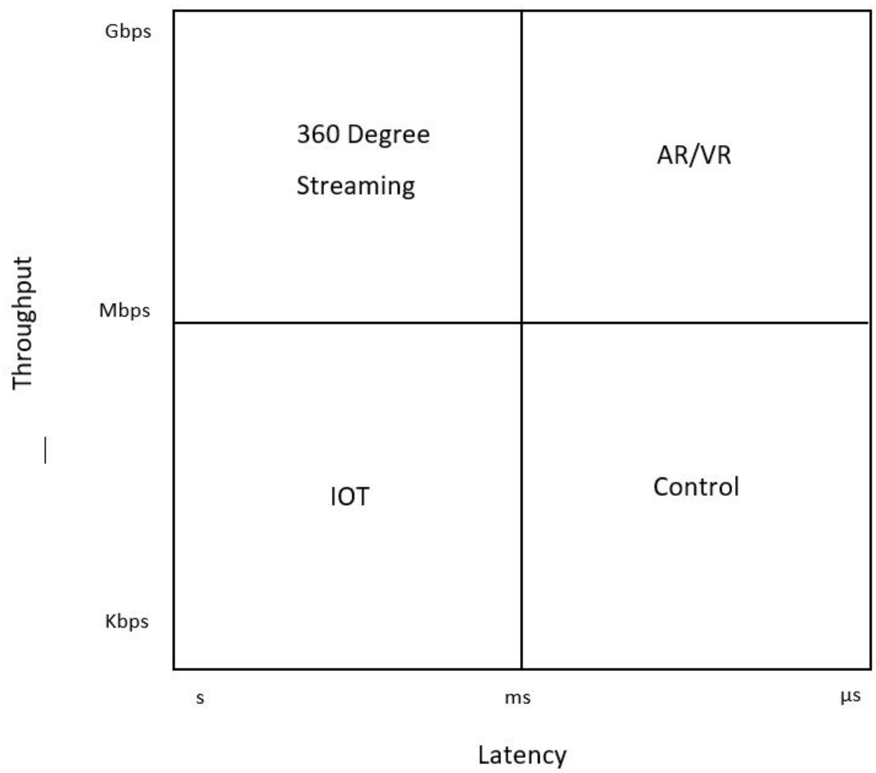

Figure 2: Bandwidth-to-Latency Relations

Assumptions and Limitations

Assumptions

- Environmental conditions in the lab room will remain nominal for electronics.
- Internet access will be sufficient and reliable.
- Outside wireless signal interference will be negligible.
- Users will be students and researchers so they will be somewhat familiar with testbeds.
- Users will participate in reporting bugs if or when they arise for continued improvement.
- ISU Electrical and Computer Engineering department will publish or advertise that this testbed is ready for students and researchers to use.

Limitations

- All software used must be open-source.
- Lab room has space for only 110 total nodes.

- Budget is a factor so the amount of software-defined radios (SDRs) is limited.
- Two semesters is the maximum time to spend designing and developing this project.
- Some elements of this project are new to us so time will need to be spend researching.

Expected End Product

The end product for this project will be a fully operational wireless communication system testbed. The testbed will be comprised of a grid of individual nodes equipped with various communication system hardware and software platforms. These nodes will be monitored and controlled by a server that can be access via a web platform that we will implanenting.

Related Work / Literature Review

We researched several existing wireless testbeds in our project's initial stages to define what a testbed is and how its elements must perform together to comprise a system. One such platform is Powder [2] located in Salt Lake City, Utah. Powder's goal is to provide a wireless testing environment that spans city-wide areas including a downtown area, a residential neighborhood, and the University of Utah campus. Researchers are able to build mobile wireless networks using 4G and MIMO technologies. Our CyWi project will be taking it one step further by introducing 5G technologies with the OpenAirInterface [1] platform. One Powder testbed cluster is built using the well-known, open-source Emulab [3] environment. We plan on using Emulab to streamline development and provide a standardized platform for our users.

Orbit [4] is another wireless testbed we researched. It is managed by several universities in the New York and New Jersey region and has been around since 2005. The project hosts multiple testbeds of various sizes from 2x2 radio node grids up to a 20x20 grid. The CyWi lab will start out with a maximum of 11x10 nodes. Both Powder and Orbit have the same experiment lifecycle including specifying the architecture, parsing the specs to a server, allocating resources, configuring the nodes, and gathering logs. The CyWi lab will have the same general functions. However, our focus will be on providing a platform specifically for IoT and 5G experimentation.

Acknowledgements

The CyWi team would like to give special thanks to professor Professor Hongwei Zhang . His technical assistance on the project will serve to be invaluable to us. Additionally we would like to thank Professor HongWei Zhang for securing the funding to make this project possible. Finally we would like to acknowledge the Iowa State Department of Electrical and Computer Engineering for providing the lab space in Coover Hall to create the CyWi testbed, and also we would like to thank ETG for their assistance in wiring the testbed.

III. Proposed Approach and Statement of Work

Proposed Approach

High-Level Block Diagram

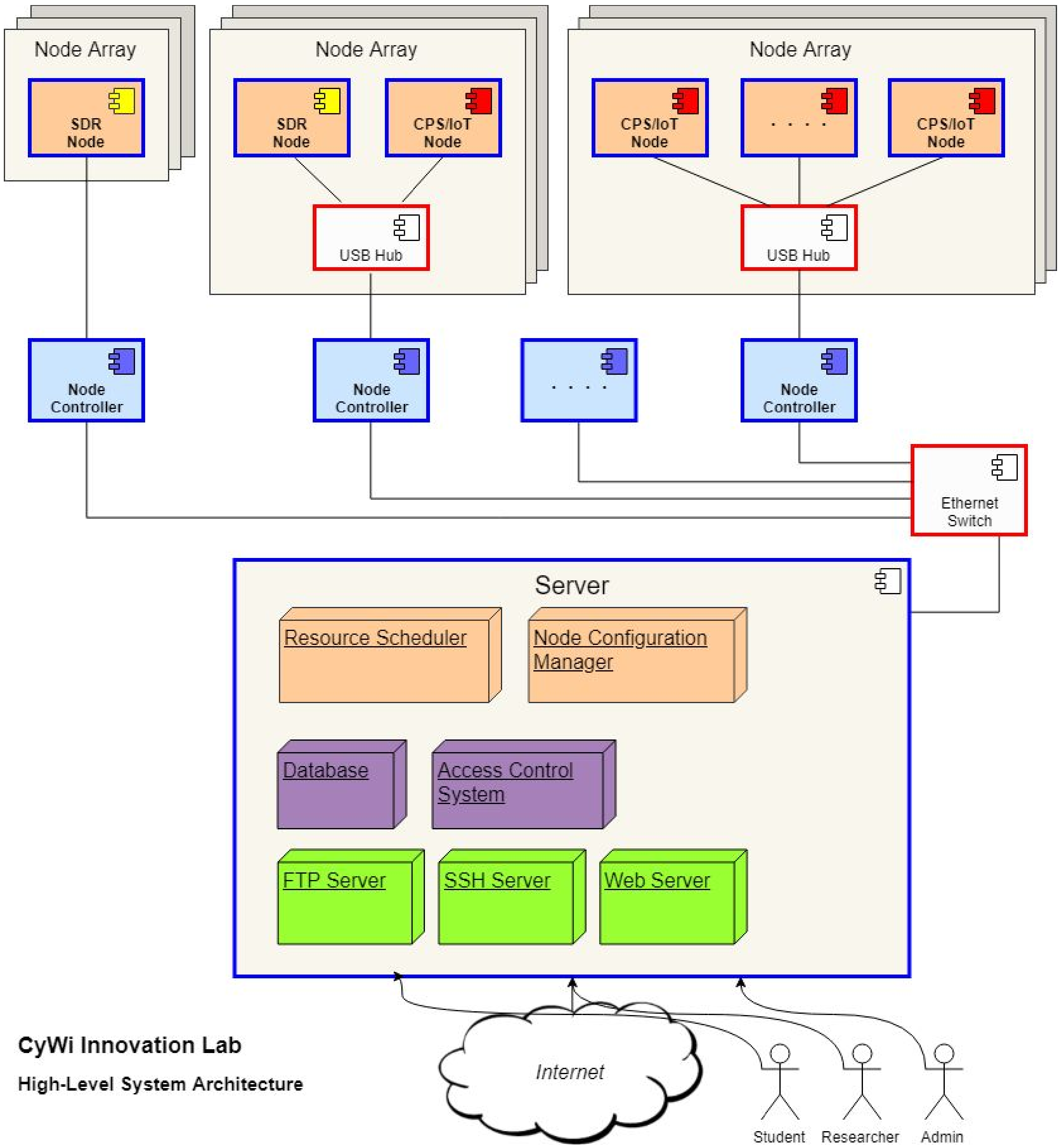

Figure 3: High-Level Block Diagram

Users will connect to the testbed via a Web interface to configure their experiments. The server, which contains the resource scheduler and node configuration manager components, will provide the node controllers (Intel NUC mini-PCs) with experiment specs and configurations. The node controllers will, in turn, pass configurations on to the necessary nodes. Users will run one of three general types of experiments: SDR only, IoT motes only, and a combination of SDR and IoT motes. The SDR and IoT motes are connected to the node controller via USB. After the experiments have finished, experiment data can be exported via FTP.

The CyWi Innovation Lab will satisfy the following functional and non-functional requirements:

Functional Requirements

The CyWi testbed is required to perform the following functions:

- Administrative users can create user accounts.
- Allow user access via web interface and SSH.
- Securely store user profiles, experiments, and usage statistics in a database.
- Track resource availability.
- Reserve resources as requested by users.
- Create, configure, and run experiments.
- Allow the user to export experiment data.

Constraints Considerations

Restrictions that define the project limitations are:

- Lab room ceiling tiles are arranged in a grid of 11x10, which limits our maximum quantity of CPS/IoT devices to 110 CPS/IoT development kits.
- Powerful and configurable wireless devices, such as SDRs, are expensive. High costs limit the quantity of devices available to us and decreases the size of the testbed.
- Scheduling a time for six active students to meet is challenging and strains our ability to work together for longer periods of time.

Technology Considerations

One of the chief motivations for building the CyWi lab is to allow students and researchers to experiment with up-and-coming technologies. Therefore, the hardware selected for the testbed must be capable of supporting multiple modern standards such as 5G, SmartAg, CPS, IoT, and AR/VR. Finding one type of device that covers all of these would be difficult and expensive so we chose to focus on two classes of devices: software-defined radios (SDRs) and CPS/IoT development kits.

Including SDRs in the testbed allows us to offer our users a high level of control. An SDR's FPGA is fully customizable which allows researchers to have direct control over signal processing, removing the need for additional radio equipment. Since the testbed is designed to be used entirely remotely, this seems like the most elegant solution. The SDRs will be useful for experiments that require high bandwidth and low latency such as AR/VR-based models.

CPS and IoT have less stringent requirements when it comes to bandwidth and latency so a better option is to install a large number of affordable development kits with built-in support for Wi-Fi, Bluetooth Low Energy, and Zigbee. The next decade will see an explosion of CPS/IoT development so students and researchers will benefit from experimenting with wireless communication protocols designed for these applications.

Accessibility is another chief motivation of the CyWi lab and nothing is more accessible than open-source software. Not all development devices support open-source development software so this was a major consideration when deciding on which testlab devices to purchase. The Texas Instruments development kits support a variety of open-source real time operating systems (RTOS). For cellular research, we settled on Ettus USRP SDRs and the OpenAirInterface platform. The OpenAirInterface community is continuing to develop LTE and 5G solutions in an open environment.

Technical Approach Considerations

The CyWi team is approaching this large project with utmost seriousness. This testlab will remain at Iowa State University long after we have all graduated, and the return on investment for the funding that made it possible will be expressed in years of student and researcher hands-on experience.

Most of the team must undergo a steep learning curve when it comes to wireless technologies. For this reason, we are spending the first two months surveying CPS/IoT platforms and studying wireless technologies. Once we have a better understanding of the technology, we will be in a position to purchase devices that best suit our needs. We are also experimenting with other existing wireless testbeds such as Powder and ORBIT to determine which features we would like to implement and how to set our project apart from those testbeds.

All lab hardware will be installed by Iowa State's ETG. Once they finish, we will prepare each of the Node Controllers with a Linux distro and verify that the control plane has no issues. At this point we will begin specializing. Server administrators will be in charge of setting up access control, database, and each of the Internet-facing services including the Web interface. Powder has a clean user interface and is an open-source project so we look forward to borrowing some code from them; our client has suggested this and has signed off on it.

Some SDR and CPS/IoT profiles will also need to be constructed on the nodes. A couple team members will turn their attention on the node manufacturer documentation to determine the best way to proceed. Student users will appreciate a simpler profile, but researchers will expect a wide-open profile to allow them to reprogram FPGAs, write drivers, etc.

At this point we will be ready to begin system integration. Building interfaces between each component will take time but by then we will have experience in our respective positions. Together we will define and implement APIs across system components and end up with a cohesive testbed that's ripe for further demonstrations and testing.

Testing Requirements Considerations

As we get further into the project implementation, we will be testing components as they are getting built. We must be sure that each component works individually before system integration takes place. The services connecting to the Internet will be thoroughly tested, both from on and off campus. New user accounts will be registered and updated with new information. The database queries must be properly vetted to return the correct data. The Node Configuration Manager must be able to communicate with each of the Node Controllers and make quick configuration changes. The Resource Scheduler must be aware of each node's status and never over-subscribe a node; for this to happen, a solid finite-state machine must be implemented and tested. The Node Controllers must also have a recovery mechanism in place to keep downtime to a minimum. All of these systems will be thoroughly tested promptly after being built and again after system integration. Once each of the components are combined into one testbed system, we can begin demonstrating our product.

The last three or four weeks of this project are devoted entirely to demonstrating what the testbed is capable of. During this time, we plan to create experiments that show off all our supported wireless protocols and features. Bluetooth and Zigbee applications with the CPS/IoT nodes will have many possibilities such as building a mesh network and measuring its performance. For cellular network, we will experiment with using SDRs to generate custom signals. Implementing LTE between two SDRs will prove interesting and we will attempt to use a 5G stack to show that CyWi is capable. After each experiment, the report will be exported and looked over to verify a sanity check. This final stage of the project will serve a dual-purpose: we will be demonstrating our system while at the same time discovering and troubleshooting any potential bugs.

Security Considerations

In order to physically access the testbed, individuals will have go through the approval process similar to the other labs in Coover Hall. Since data will be stored locally on the server, limited data security will be required. Additionally, individuals that wish to access the testbed remotely will have to request access to the server.

Safety Considerations

There are a few safety considerations with the physical construction of the testbed. The testbed nodes will be hung from the ceiling and there will be a lot of electrical wiring involved with the testbed. Proper care will have to be taken to ensure that our hardware does not pose any risks to individuals using the testbed. To ensure proper installation of the electrical wiring and hardware, ETG will be contracted to perform the installation.

Possible Risks and Risk Management

With the proposed size of the testbed, when fully functional, it will require a considerable amount of power. Additionally there will be a lot wiring entwined into the testbed. There poses a risk for electric shock from potentially misusing/handling the hardware. In order to mitigate this risk the testbed is mounted to the ceiling, all live wiring will be properly shielded and out of reach.

Project Proposed Milestones and Evaluation Criteria

Milestone 1: Installation of all the hardware

This milestone will be composed of ordering all of the SDRs, TI products, and NUCs and installing them. This milestone will be considered complete if we have acquired all of the products and they are wired together in the designated room in Coover. This will be accomplished by March 22nd

Milestone 2: Completed Software

This will be the bulk of this project and it will include developing all of the necessary software for our test bed to run. This will include configuring all of the nodes and node controllers, implementing the node configuration manager, implementing the resource scheduler, and the web server. This milestone will be considered completed when the test bed is up and running and researchers are able to run experiments through it. This will be completed by December 6th.

Milestone 3: Developing Test and Writing Article

This milestone is for the purpose of showing that the test bed does indeed work and documenting the whole process. As we are developing this project we will be documenting the path we took and writing an article that will display the journey of making this test bed. We will also spend a month developing our own experiment that we will run through the test bed. This milestone will be considered completed with a documented article and a completed experiment that has been run. It will be completed by December 13th.

Project Tracking Procedures

We will track our progress by following the guidelines of our gantt charts. We have a gantt chart for the first semester and the second semester which are shown below.

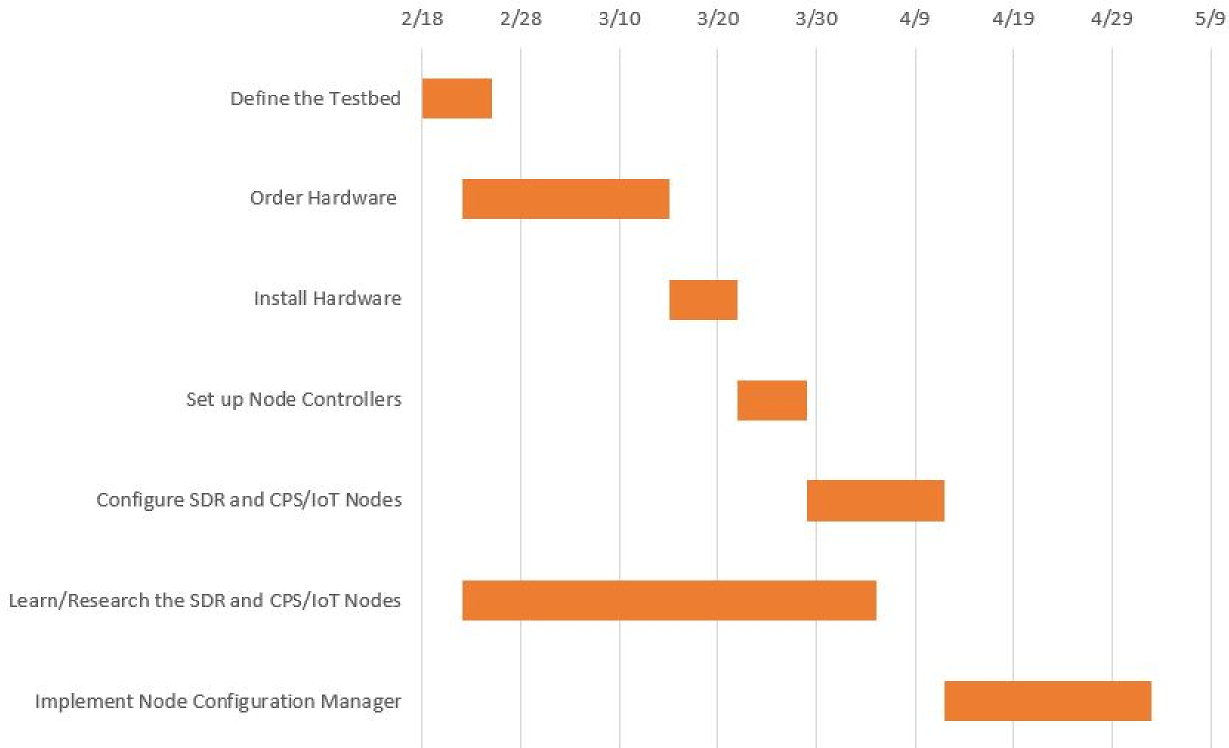

Figure 4: Semester 1 Gantt Chart

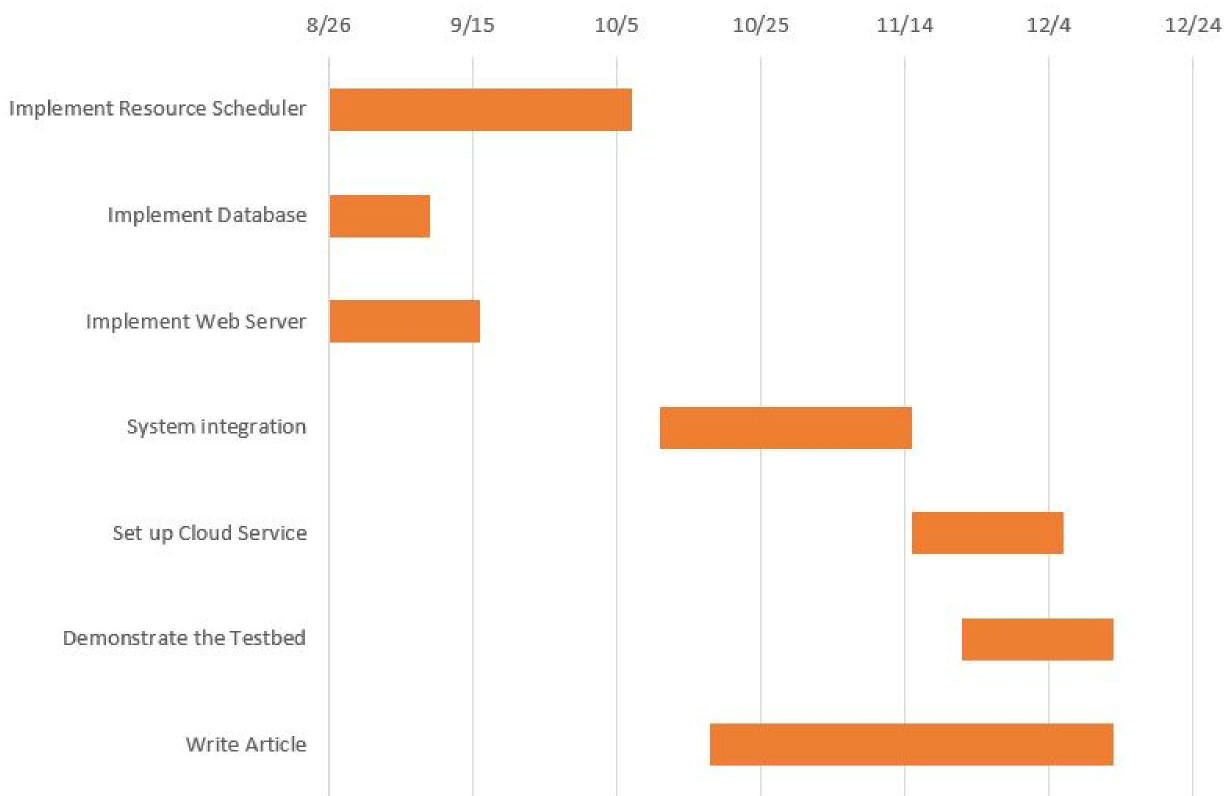

Figure 5: Semester 2 Gantt Chart

Statement of Work

Number and describe for each task that needs to be completed, 1) the task objective, 2) the task approach, and 3) the expected task results.

Define the testbed

- Approach: Researching technical needs of the testbed.
- Results: Determines what we need to order for hardware.

Order Hardware

- Approach: Communicate with professor as to what hardware we need.
- Results: To make sure we will be receiving the hardware we need in a timely manner

Install Hardware

- Approach: Assembling hardware in the lab.
- Results: Sets up the bigger part of the testbed.

Set up Node Controllers

- Approach: Determining what will make these nodes configurable to users.
- Results: The controllers will be responsible for reporting status and managing each node.

Configure SDR and CPS/IOT Nodes

- Approach: Determine how we need to set them up to make them configurable to the users.
- Results: Sets Nodes up for experimentation.

Learn/Research the SDR and CPS/IOT Nodes

- Results: We will know more about what configuration options to allow.

Implement Node Configuration Manager

- Approach:Using what we learned before, we make the list of FPGA and network layer configurations that will be available and write code that will set them up.
- Results: Nodes will be configurable.

Implement Resource Scheduler

- Approach: Use of scheduling algorithms, determining how it handles users requests whether it accepts or denies.
- Results: Manages which nodes are available for checkout, and to who.

Implement Database

- Approach: Determine how information will be stored and then made available to the user when an experiment is done.
- Results: We will have a database from which to pull experiment data from.

Implement Web Server

- Approach: Software for file transfer. Front End Design: software for SSH process and Web interface.
- Results: We will have a way of being able to move experiments onto the testbed.

System Integration

- Approach: Testing to make sure our different components work together.
- Results: Users will be able to put files on the testbed and get test results.

Set Up Cloud Service

- Approach: Determining how everyone will be able to get access to the testbed.
- Results: We will have a way for everyone to easily use the testbed

Demonstrate the Testbed

- Results: Shows that we have a functioning project

IV. Estimated Resources and Project Timeline

Personnel Effort Requirements

This section is just a draft of the early stage of the project. It will definitely get updated when we have a better idea on how long each of these tasks will take.

Table 1, seen below, includes the major project tasks with descriptions and also the estimated hours for each task.

Task	Description	Estimated Hours
Weekly Report	Record past week's work and compile them into a report.	25 hours
Project Plan	Create a project plan and keep it updated throughout the project.	20 hours
Define the Testbed	Define exactly what the user experience will be and how these systems will fit together. Define explicit use cases.	10 hours
Order Hardware	Purchase SDRs, TI development kits, Intel NUC8s, server, and necessary networking equipment.	5 hours
Install Hardware	Physically install and connect all the devices.	10 hours
Set up Node Computers	Install Linux distro on each node computer and verify that they communicate with their wireless devices.	15 hours
Configure Wireless Devices	Install open-source software on each SDR and TI development kit as well as configure them on the private network.	30 hours
Learn/Research the SDR and	Learn how to use each of	100 hours

TI Devices	these technologies.	
Implement Controller Server	Implement the controller software on the server to allow it to configure the node computers. The controller server should be able to send files and commands to individual nodes. The node computers, in turn, will configure the wireless devices.	50 hours
Implement Resource Scheduler	Implement the resource scheduler to grant resources (access to certain nodes/wireless devices) to users per constraints.	100 hours
Implement Database Server	Implement a database on the server to store multiple tables for things such as user authentication, user statistics, node/wireless device status/availability, schedule reservation charts (if needed), experiment data.	30 hours
Implement Front-End Server	Create a launch site for researchers to interact with so they can login, request resources for a given time period, and gain access to the nodes/wireless devices.	50 hours
Interface Front-End, Resource Scheduler, Database, and Controller Servers (System Integration)	Implement interfaces between the servers so they can interact. Users should be able to login, query databases, checkout resources, configure wireless devices, run experiments, and export experiment data.	100 hours
Set up Cloud Service	Implement the cloud service	30 hours

	so that users can remotely access the testbed.	
Demonstrate the Testbed	Run some different kinds of experiments on the 'finished' testbed to demonstrate what it's capable of.	30 hours
Write Article	Article about how the testbed was created and how it works, etc. Possibly publish. Why it was made? Benefits? Audience: 1) users 2) developers	30 hours

Table 1: Major Tasks with Description and Estimated Hours

Other Resource Requirements

Every team member needs to have a computer that have a text editor with certain language compiler. To be able to test our user interface in the future, we need to have access to the internet. Besides that, we need access to the testbed lab in order to configure wired and wireless devices such as radios and server. And also, in the early stage of our project, we plan to take a more straightforward approach, by limiting most tests to be in wired connection. Furthermore, in order to further understand the nature of a wireless communication testbed, we need to register an account for access.

Financial Requirements

Total Budget	Undetermined
Cost of Hardware	**Price ($)**
2 x USRP B210	2 x 1,216 = 2,432
20 x TI CC26x2R	20 x 39.99 = 799.80
1 x Server Workstation	1,500
Total	4,731.80

Table 2: Expected Cost of the Project

The total budget of the project is currently undetermined because we have to check it with our academic advisor. The project is fully funded by our professor, Professor Hongwei Zhang. The cost of the project is only made up of the hardware to power the testbed.

The project requires a few radios, each with different capabilities, such that the testbed can support as many experiments as possible. Moreover, it also requires a few computing devices which can run all the radios.

The external tools that we plan to use are all open-source. For example, OpenAirInterface [5] is created by EURECOM, which can be used by the community, and Powder [6] is created by University of Utah under the sponsor of National Science Foundation. Since we are planning to develop the code on our own and build upon open-source code, we expect to not spend any amount on software for this project.

Project Timeline

Name	Description	Timeframe	Start Date	End Date
Define the Testbed	Define exactly what the user experience will be and how these systems will fit together. Define explicit use cases.	1 week	2/18/2019	2/25/2019
Order Hardware (hardware & software platform research)	Purchase SDRs, TI development kits, Intel NUC8s, server, and necessary networking equipment.	3 weeks	2/22/2019	3/15/2019
Install Hardware	Physically install and connect all the devices.	1 week	3/15/2019	3/22/2019
Set up Node Computers	Install Linux distro on each node computer and verify that they communicate with their wireless devices.	1 week	3/22/2019	3/29/2019
Configure Wireless Devices	Install open-source software on each SDR and TI development kit as well as configure them on the private network.	2 weeks	3/29/2019	4/12/2019
Learn/Research	Learn how to use each of	6 weeks	2/22/2019	4/5/2019

the SDR and TI Devices	these technologies.			
Implement Controller Server	Implement the controller software on the server to allow it to configure the node computers. The controller server should be able to send files and commands to individual nodes. The node computers, in turn, will configure the wireless devices.	3 weeks	4/12/2019	5/3/2019
Implement Database Server	Implement a database on the server to store multiple tables for things such as user authentication, user statistics, node/wireless device status/availability, schedule reservation charts (if needed), experiment data.	2 weeks	8/26/2019	9/13/2019
Implement Front-End Server	Create a launch site for researchers to interact with so they can login, request resources for a given time period, and gain access to the nodes/wireless devices.	3 weeks	8/26/2019	9/20/2019
Implement Resource Scheduler	Implement the resource scheduler to grant resources (access to certain nodes/wireless devices) to users per constraints.	5 weeks	8/26/2019	10/11/2019
Interface Front-End, Resource Scheduler, Database, and Controller Servers (System integration)	Implement interfaces between the servers so they can interact. Users should be able to login, query databases, checkout resources, configure wireless devices, run experiments, and export experiment data.	5 weeks	10/11/2019	11/15/2019
Set up Cloud	Implement the cloud service	3 weeks	11/15/2019	12/6/2019

Service	so that users can remotely access the testbed.			
Demonstrate the Testbed	Run some different kinds of experiments on the 'finished' testbed to demonstrate what it's capable of.	3 weeks	11/22/2019	12/13/2019
Write Article	Article about how the testbed was created and how it works, etc. Possibly publish. Why it was made? Benefits? Audience: 1) users 2) developers	2 months	10/18/2019	12/13/2019

Table 3: Project Checkpoints with Descriptions and Estimated Deadlines

The table above shows the overall timeline that can be broken down into **monthly** deadlines:

February
- The testbed should be clearly defined with the agreement between the students and the client(s)

March
- Required hardwares are ready to be installed in the testbed
- Each node is filled with its desired computer

April
- The functionality of SDR/TI machines are thoroughly studied
- Connections between each nodes are established

May
- Controller Server is readily to be deployed into the system
- Summer Break

August
- Semester II starts

September
- Database Server & Front-End Server are ready to be implemented

October
- Resource Scheduler is ready to be implemented

November
- Interface platform is created for the communication between Controller Server, Resource Scheduler, Database Server, and Front-End Server

December

- Finish Cloud Service setup
- Testbed is ready for demonstration

These are the major deadlines to be fulfilled throughout the project year. Alternatively, *Figure 4* and *Figure 5* can also be used as visual references for a brief timeline.

V. Closure Material

Closing Summary

Technological innovation has been steadily increasing over the last few decades. The future is fast approaching and Cyber-Physical Systems, the Internet of Things, and 5G wireless will soon be mainstream. These highly disruptive technologies will change every facet of modern life including healthcare, home safety, industrial automation, transportation, education, social connectivity, entertainment, and more.

Such a social seachange cannot be made possible without the continued efforts of researchers. The Iowa State University CyWi innovation lab aims to provide students and researchers the resources to learn about emerging technologies and to push the boundaries of possibility forward. Registered users will have a well-documented matrix of SDR and CPT/IoT nodes at their disposal with the ability to run custom experiments across several wireless protocols. Exportable experiment results will allow researchers to take their data offline for further analysis. Innovating in such a lab environment will speed up the next technological revolution.

References

1. https://www.openairinterface.org/
2. https://powderwireless.net/
3. https://www.emulab.net/portal/frontpage.php
4. https://www.orbit-lab.org/
5. https://www.ettus.com/product/details/X310-KIT
6. https://www.ettus.com/product/details/UB210-KIT
7. http://www.ti.com/tool/launchxl-cc26x2r1
8. https://www.intel.com/content/www/us/en/products/boards-kits/nuc/kits/nuc8i7hvk.html