Leveraging Industry 4.0 in Education for Remote Implementation in a Team-Based Computer Engineering Capstone Project

Sangit Sasidhar
Department of Electrical and Computer Engineering
National University of Singapore
Singapore
sangit@nus.edu.sg

Jithin Vachery
School of Computing
National University of Singapore
Singapore
jithin@comp.nus.edu.sg

***Abstract*—This Innovative Practice Category Full Paper presents the remote implementation of an embedded systems capstone project for computer engineering students. A capstone project is a feature of most undergraduate programs in computer engineering. Such a project is usually meant to expose students to the development of a large system from conceptualization to its final implementation, involving substantial design and development of hardware and software components.**

In our university, students were given the opportunity to work on a cutting-edge problem focused on healthcare - "Designing a wearable device that automatically detects human activities". It is an area where a large amount of research is ongoing, and hundreds of scientific papers are published every year. In addition, students had to evaluate and adopt techniques from existing literature and adapt them to meet the problem requirements. Students equipped themselves with state-of-the-art hardware, Biosignal processing, machine learning, power optimization and secure communications to design the wearable. Thus, the project reinforced their knowledge of fundamentals, while exposing them to a problem with no obvious solution. Through the capstone project, students are able to better appreciate the relevance of the various components in the computer engineering curriculum to large-scale computer engineering projects. Students are organized into teams of six to execute the project.

The COVID-19 pandemic resulted in the university migrating the teaching online for all courses. This was particularly challenging to implement for the capstone project as one of the key requirements is for the members to work together and subsystems to interact with each other. A course refresh resulted in a credits update and this provided a unique opportunity for the teaching team to redesign the project with core Industry 4.0 technologies such as hardware acceleration, remote processing, networking, remote analytics, and secure protocols. This work presents a framework for its implementation, reviews the challenges encountered and the processes put in place to ensure data security and smooth running of the project in the event of future disruptions. Quantitative and qualitative results from the course feedback surveys are analyzed to gauge student response to the implementation and compared to the previous version.

***Keywords*—Capstone Project, Remote Learning, Industry 4.0.**

I. Introduction

Project-based learning (PBL) is an effective way to bridge the gap between theoretical and hands-on knowledge [1] by context-specific learning, motivating students to become active learners and achieving common goals by peer learning and sharing [2]. The context-specific learning is achieved using authentic problems that are informed by real-world practices and issues [3]. A well-designed project covers the six levels of Bloom's taxonomy [4] and improves critical thinking along with developing soft skills [5]. There are multiple studies that present the effectiveness of PBL, including but not limited to software engineering [6], bio-mechanics [7], electrical engineering [8], and analytical design [1].

A capstone project is a feature of most undergraduate programs in computer engineering. Marin et al. describe it as "the crowning achievement in a student's academic curriculum, which integrates the principles, concepts, and techniques explored in earlier engineering courses" [9]. A capstone project combines PBL with an experiential learning activity that requires students to perform product design, development, testing, and documenting, in a single team-based project [10]. Accreditation Board for Engineering and Technology (ABET) Criterion 4 [11] specifies that "Students must be prepared for engineering practice through a curriculum culminating in a major design experience based on the knowledge and skills acquired in earlier course work and incorporating appropriate engineering standards and multiple realistic constraints". The scope of the selected projects should enable the students to execute them within the duration of the course, utilizing the skills, knowledge, and practices learned in foundational courses [12].

In computer engineering programs, such a project is usually meant to expose students to the development of a large system under the Conceive, Design, Implement, and Operate (CDIO) framework [13] from conceptualization to its final implementation and operation, and involves substantial design and development of hardware and software components. The importance of the capstone project in introducing students to engineering design is well documented in literature as it better prepares graduates for engineering practice [14]. In addition, analytical and problem-solving skills that are key for employability in the industry are often honed through work done in engineering capstone projects [15].

The emergence of Industry 4.0 (I4.0) requires the modern student to be equipped for the digitized industry and the corresponding associated challenges. Technologies such as the Internet of Things (IoT), robotics, virtual reality (VR), secure communications, distributed systems, augmented reality (AR), artificial intelligence (AI), and cloud computing form the building blocks for designing smart factories that can achieve high efficiencies and increased productivity [16]. Ritter explores the challenges in training students in I4.0 and the need for educators to adapt by understanding contextual insights to fulfill the needs of the modern industry [17]. Patricia et.al. compared the experiential and challenge-based learning experiences with industry at two universities and its implications for practices and the corresponding challenges [18]. Researchers in [19] address the alignment of engineering education with the needs of the industry through a capstone-oriented course wherein the students were tasked to design IoT systems to solve a university parking problem. Given the complexities involved in students selecting a suitable project, and the difficulty faced in fairly evaluating projects of disparate complexities [12], a fixed theme was selected so as to minimise the adverse impact of the challenges listed. The theme focused on exposing students to various I4.0 technologies such as machine learning and AI, remote analytics, secure communications, and IoT. Students are given substantial freedom in deciding the overall system architecture, including hardware, software, as well as form factor aspects.

The migration to online learning (OL) due to the different isolation and lockdown restrictions of COVID-19 has presented numerous challenges for engineering education. The transition to OL has been mild on subjects such as software engineering and computer science with an emphasis on an interactive and experience-based environment [20]. The authors at [21] explore the effect of the pandemic on engineering education, especially the absence of hands-on experience from traditional labs, and suggest the need to incorporate artificial intelligence and virtual reality elements to innovate them. Svatos et al. address the challenges of online teaching of practical classes under these restrictions utilising a combination of hardware boards and software applications for an electronic class [22]. Authors of [23] explored the benefits of capstone projects based on existing IoT devices on student engagement during the pandemic. This paper explores the development and implementation of an embedded systems capstone project remotely.

The rest of the paper is organized as follows: Section II gives the background and details of the capstone course, and Section III details the capstone design process. Details of student performance in the course and teh corresponding discussions are given in Section IV. The paper concludes with Section V.

II. Background

A. The Capstone course

The capstone project in its current iteration is a semester-long, 8 Credits (1 Credit implies an investment of 2.5 hours per week) course, usually taken by third-year computer engineering students. The computer engineering program in our university is multidisciplinary, wherein students belong to both the Department of Electrical and Computer Engineering and the Department of Computer Science. The first four semesters mostly involve core courses, and students have the option to take the capstone course in their fifth or sixth semester. Students have to go through the core courses such as digital fundamentals, ARM Cortex M3 programming and interfacing, real-time operating systems, and software engineering, before taking this course. The course is co-taught by 2 to 3 lecturers, at least one from each department. The course is conducted using a combination of lectures, design walk-through sessions, and laboratory sessions over 13 weeks. Students are organized into teams of 6 to execute the projects. Through the capstone project, students are able to better appreciate the relevance of the various components in the computer engineering curriculum to large-scale computer engineering systems.

B. Learning Outcomes

The course learning outcomes (CLOs) of the capstone project are that students should be

1) confident in managing large-scale systems
2) confident in developing embedded systems that integrate hardware, communications, and software
3) able to estimate the time for a complex project

4) aware of security, privacy, and potential societal impacts of a system
5) able to work in a team collaboratively

The course takes the students through product conceptualization, requirements capture, application design, hardware/software partitioning, hardware development, software development, system integration and verification, and system commissioning. Students learn from a hands-on project to design and build an embedded system. Emphasis was also made on systematic software engineering practices, as it is key to rendering students ready for the industry [24].

III. Capstone Project Design

There were several considerations when choosing a capstone project theme. Cheville et al. [25] identified some attributes of successful capstone design projects, including "being viewed as worthwhile", being related to the field of specialization in engineering, and involving "modern and emerging technologies with which most of the students would have some familiarity". Service learning projects with a potential social impact were more likely to be viewed as worthwhile by students [26]. State-of-the-art problems of a reasonably complex nature with no obvious solution yet are intrinsically exciting while being centred in the computer engineering discipline; thus, they motivate students to read widely and build products that may have a real societal impact. Adding a competition element to the evaluation maintains high student engagement and interest, and a fixed project addresses the challenges associated with project selection and allows for fair evaluation. With these considerations in mind, the scope of the project was defined.

A. Project Overview

Wearables provide a way to detect human activity automatically and have been used in wide-ranging scenarios. Fitness tracking is likely the most prevalent use of activity detection [27]. Using wearables such as smartwatches or fitness trackers, users can monitor their step count and more advanced sports training such as stride and pace for running, strike force for tennis, stance for basketball, etc. Activity detection has also been used in medical scenarios, such as detecting falls [28] or anomalous movements that indicate medical conditions [29]. Activity detection has been applied in our daily lives to infer our minute-by-minute activities at home, during a commute, at work or school, while shopping, exercising, etc. In this project, the goal is simple:

Designing a wearable device that detects human activities automatically

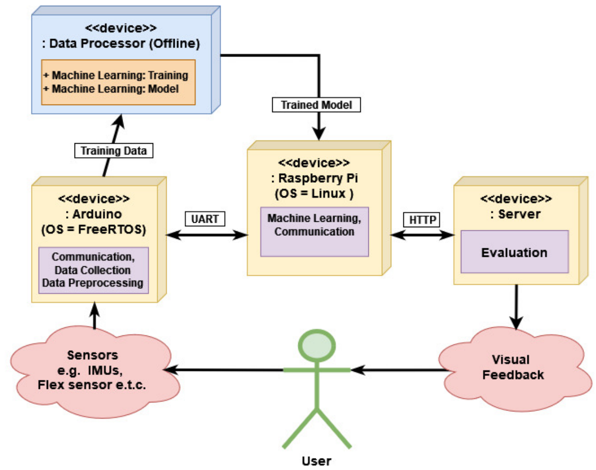

Fig. 1: Version A of the Capstone Project

Sub-System	Student A	Student B
Hardware	Sensor Integration	Power Management
Software AI	AI Algorithm A	AI Algorithm B
Firmware	FreeRTOS / Arduino	Server Communication

TABLE I: Version A - Subsystems and student roles

B. Version A

The students were randomly grouped into teams of six. Each team was given a Raspberry Pi (RPi) single-board computer (SBC) and an Arduino Mega microcontroller board. A set of standard components, including Inertial measurement units (IMU), level shifters, and other basic ICs, were provided. Each team could also purchase additional components/materials subject to a certain budget limit. The system had to be battery-operated, so students were encouraged to use power judiciously. The project broadly had three primary subsystems, namely hardware, firmware, and software AI, each handled by two students implementing different methodologies as shown in Table I

The overall system design is shown in Fig. 1, and the teams were given the following guidelines:

- Wearable: The device must be worn by the human performing the activities. All sensing and compute must be done on this wearable. Weight, form factor, and comfort are all potential evaluation criteria.
- Wearable automatically detects these dance moves and should work when worn by anyone
- By performing the specific activities, the wearable detects the dance move and communicates the detected dance move to the class server in a secure manner
- Server will indicate a random dance move, then the individual will perform the dance move with the wearable system, and the system will be evaluated

to see if it has correctly recognized the specific dance move.

This version of the capstone project was a semester-long 6-credit course and successfully ran for 6 iterations. An internal review process at the university led to a curriculum renewal, and the computer engineering capstone project was updated from 6 credits to 8 credits. With this update in credits, there was a requirement to refresh the capstone project. The advent of the COVID-19 pandemic and the corresponding shift to online teaching coincided with this credits refresh that led to the redesign of the capstone project.

C. *Version B*

The primary objective of Version B was similar to that of Version A with the following additional guidelines:

- A wearable device and system to be designed to coach the dancers as they perfect and synchronize the dance moves.
- Three dancers will move into the specified relative positions and then perform each specific dance move shown on the server screen.
- Wearable automatically detects these dance moves, the dancers' positions, and how synchronized they are with each other, even when the three dancers are dancing in separate physical locations.
- The system should be realized with the hardware platforms provided and accessible remotely.

The key components of the Version B of the capstone project (Fig. 2) are detailed below:

*1) **Hardware Platforms**:* The increase in credits and the adoption of I4.0 technologies for remote implementation and operation resulted in a team of six students implementing six sub-systems that had different hardware requirements as compared to Version A. Each of the teams was given the following hardware platforms:

- Ultra96-v2: An ARM-based board with integrated Field Programmable Gate Arrays (FPGA) used for hardware acceleration. The Ultra-96 board is housed in the maker-space and installed with PYNQ Linux. PYNQ provides a Jupyter-based framework with Python for embedded system design on the Ultra96.
- Beetle BLE: An Arduino Uno board with integrated Bluetooth low energy (BLE).
- A set of standard sensors.

In addition to the above, the students used relay systems (either their own laptops or borrowed Raspberry Pis) and their own laptops for remote access. The teams could also purchase additional components/materials within their budget. The various sub-systems' dependence on different hardware platforms is detailed in Table II. It can be seen that the different sub-systems are tightly integrated, and each sub-system needs to communicate with multiple hardware platforms.

Sub-System	**Beetle**	**Ultra96**	**Relay System**	**User Laptop**
Hardware Sensors	Yes	No	Yes	No
Hardware AI	No	Yes	No	Yes
Software AI	Yes	Yes	No	Yes
Software Dashboard	No	Yes	No	Yes
Internal Communications	Yes	No	Yes	No
External Communications	No	Yes	Yes	Yes

TABLE II: Version B - Dependence of the different sub-systems on the hardware platforms

*2) **Evaluation Server**:* The evaluation server is a grading server that remotely measures the accuracy and response time of the system as follows:

- The server displays the action to be performed by the students via video conferencing.
- The students perform the action (dance move) on seeing the action prompt. This generates sensor readings, which are sent to the Ultra96 via the relay system.
- Ultra96 will identify the action/move and send the detected move to the evaluation server and dashboard. The evaluation server uses the ground truth to calculate the accuracy.

*3) **External Communication**:* External communication refers to all the communication carried over a Transmission Control Protocol/Internet Protocol (TCP/IP) network, which may be over the Internet. External communication is an integral part of any distributed system. With the ability to communicate, it is possible to coordinate and control the diverse components spread across various locations. In this project, communication needs to be established between Ultra96 and the relay system, Ultra96 and dashboard, and Ultra96 and the evaluation server. Communication between all these components uses TCP client-server architecture. The project requirements are designed to expose students to various security and connectivity paradigms.

(a) *Ultra96 and evaluation server*: Ultra96 and the evaluation server are placed in the same subnet and can reach each other over IP. Ultra96 sends and receives data from the evaluation server using a TCP server. The communication is secured via Advanced Encryption Standard (AES). The passkey is different for each team. This requirement exposes the students to the need for encrypting their

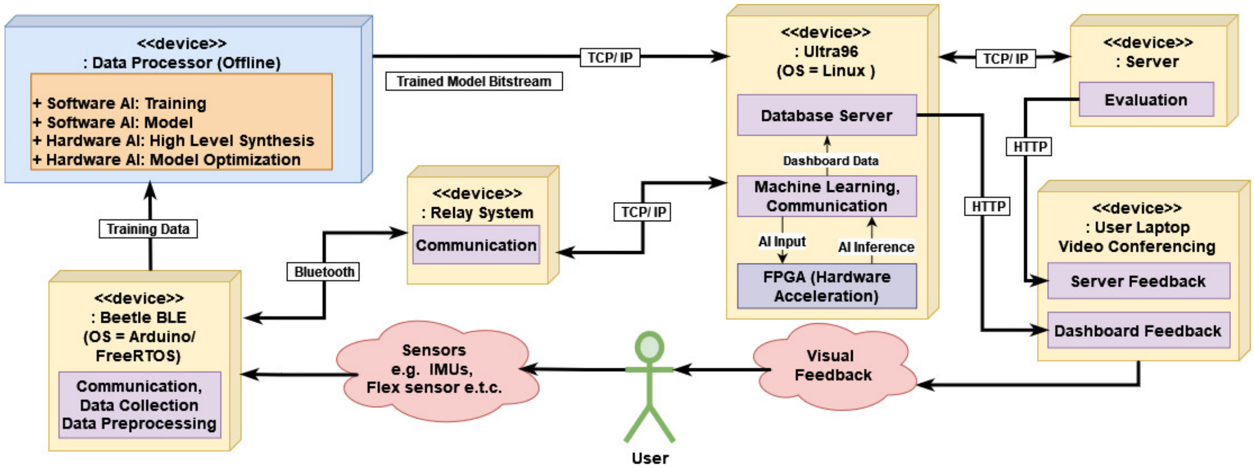

Fig. 2: Version B of the Capstone Project

data. This is particularly important to avoid false penalization. Without encryption, a team may accidentally register as a different team and could have a poor performance recorded. Additionally, it is demonstrated to the students how someone may intentionally sabotage their team's performance by falsifying their identity.

(b) *Ultra96 and dashboard*: The dashboard provides real-time diagnostics and displays the system's performance. To this end, the teams host a web server on Ultra96, which may be connected from any location. Because of firewall restrictions and the fact that Ultra96 does not have a public IP address, the teams are required to connect to the university network using a virtual private network (VPN) before connecting to the dashboard. This exposes the students to the overall network design and concepts of public and private networks.

(c) *Ultra96 and user laptop*: Students access all the servers used in the project via their personal laptops. This laptop is also the development environment for the project source code. Students need to transfer the source files to Ultra96 and remotely execute them. Both of these are performed via Secure Shell Protocol (SSH). Additionally, the laptop connects to the evaluation server via Zoom.

(d) *Ultra96 and Relay System*: The relay systems must be connected to the Internet via Mobile Hotspot. This requirement ensures that the relay systems are outside the university network and cannot use the LAN to connect to the Ultra96. To establish a TCP connection to the Ultra96, the teams must use SSH tunneling (the use of VPN is not allowed on the relay systems).

Secure Shell Protocol (SSH): Students must connect to the university's dedicated SSH relay server and perform port forwarding. Depending on the location of Ultra96, they may be required to perform multiple port forwardings. This exposes the students to the convenience of tunneling and how a legitimate user may access servers deep inside a network. The SSH tunneling has the added advantage of being secure, which the students learn, too. Moreover, **the SSH protocol forms the heart of secure remote access for the different sub-systems via the remote shell (terminal) into Ultra96 and the ability to transfer files via secure copy (SCP)**.

In summary, the various design requirements of external communications expose the students to the different possibilities of securely connecting to resources spread across multiple networks, thus expanding the scope of making an extensive distributed system.

*4) **Internal Communication**:* One of the significant changes in the new capstone project was the need to connect all the devices wirelessly. The sensor data is gathered and wirelessly transmitted to Ultra96. This is done via Bluetooth Low Energy (BLE). This task has a key sub-problem of long-distance transmission.
Long-distance transmission: BLE is a wireless protocol used for short-distance communication, while the sensor readings must be delivered to Ultra96, which is at a different location (unreachable via BLE). This is achieved by the introduction of a "Relay System". This is a BLE-enabled device that additionally connects to the Internet. As the name suggests, the relay system relays any sensor readings received to the Ultra96 via the Internet. The system connects to and collates data from the sensors using BLE and connects to Ultra96 using a TCP connection. This makes the entire project highly distributed. The Ultra96 could be placed at any location or even virtualized. This unique project requirement exposes the student to cloud and modularized design.

*5) **Data Processor**:* The data processor is an offline system that includes the integration of the following

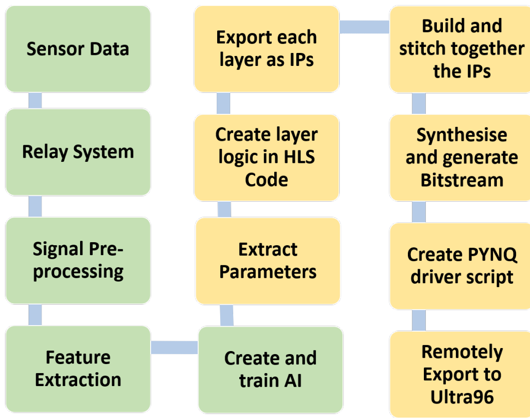

Fig. 3: The Data Processor (Offline)

sub-systems:

(a) *Software AI*: The offline processing of the activity data, feature extraction, and the training of the AI model.
(b) *Hardware AI*: Extracting parameters from the trained model, creating High-Level Synthesis (HLS) layer logic, exporting and building these layers as Intellectual Properties (IPs), synthesising and generating the bitstream and exporting it to the Ultra96 for real-time inferencing.

The software AI system accesses the sensor data remotely from the relay system and the hardware AI system exports the bitstream for hardware acceleration over TCP/IP as detailed previously. The green shaded boxes represent the Software AI, and the yellow shaded boxes represent the Hardware AI in Fig. 3.

*6) **Real-time AI**:* The real-time AI also includes the integration of the following sub-systems:

(a) *Software AI*: The real-time processing of the activity data and feature extraction.
(b) *Hardware AI*: The sub-system takes in the extracted features, feeds it to the hardware inference model on the FPGA and returns the predicted activity.

The software AI system accesses the real-time sensor data remotely from the relay system, and the hardware AI system exports the bitstream for hardware acceleration over SCP, as detailed previously.

*7) **Software Dashboard**:* The software dashboard consists of the following two functions:

(a) Offline Analytics: Displays meaningful statistics based on the analyses of the test logs and the ground truth.
(b) A real-time coaching system that streams sensor data, compares the predicted activity with the ground truth from the evaluation server and provides user feedback.

The software dashboard needs to access the processed data from the software AI system for the offline generation of player statistics. The dashboard also needs to access the sensor data stream from the Ultra96 for real-time coaching. These functions are achieved by accessing the web server (*Database Server in Fig. 2*) that is hosted on the Ultra96.

*8) **Hardware Sensors**:* The hardware sensors system primarily consists of designing the activity measurement system, including the measurement protocols for the three dancers. Each of the dancer's hardware units is linked to the corresponding relay system to transfer the data for both offline and real-time processing.

D. Support Channels

The importance of providing apt resources is outlined as a key element for the success of OL in engineering education [20]. The following support channels were provided for the course:

- Online Lectures: The introduction to different sub-systems and the corresponding theoretical content were provided as video lectures for students to view and refresh their knowledge.
- Discussion Forums: Dedicated sub-system-specific forums were initialized and monitored by the faculty and the teaching assistants. Solutions for common issues and setup problems were posted over these forums.
- Weekly Consultations: A weekly 3-hour online consultation session was setup by the different lecturers for the corresponding sub-systems to sort out the issues.
- Teaching Assistant Support: Dedicated teaching assistants (TAs) were assigned to different project teams, who would be their first point of consultation. Moreover, the individual TAs were individual sub-system matter experts for subject-specific problem-solving.

E. Evaluation Methodology

The project was evaluated continuously over the 13-week semester. The primary milestones were as follows:

(a) Overall System Design: A paper design of the high-level system architecture describing the interfaces between different subsystems
(b) Individual Subsystem: Develop the subsystems concurrently while adhering to the designed interfaces.
(c) Integrated System: The subsystems are integrated and work as a whole.
(d) Baseline systems: The integrated systems work for the baseline requirements for 1 and 3 dancers

(e) Final System: The final system works for all possible combinations of dance moves and locations. This should work for unseen dancers.
(f) Final System Report: The complete documentation of the project with students presenting their results, experiences, difficulties, and the societal impact of their project.

The final evaluation itself added an element of fun through competition, while the teams could learn from each other about the advantages and disadvantages of various implementations. All the evaluations were remote in the first iteration with complete OL, and evaluations progressively migrated to face-to-face mode with the lifting of restrictions.

IV. Results and Discussions

A. Student Feedback

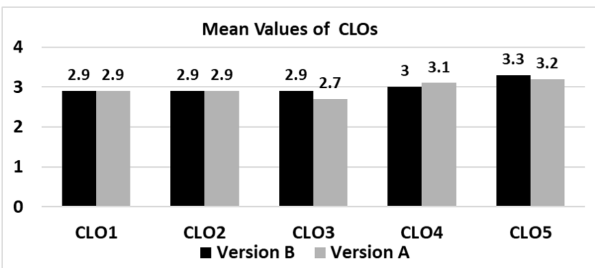

Fig. 4: Students' perception of CLO accomplishment

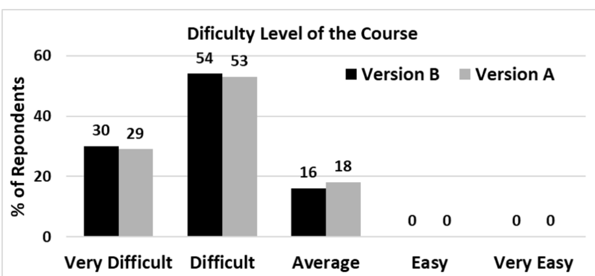

Fig. 5: Difficulty Level of the Course

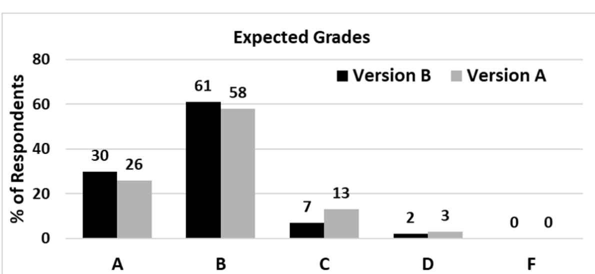

Fig. 6: Expected Grade for the Course

The Version B ran successfully for 3 iterations. The results from the formal university surveys to evaluate learning outcomes accomplishment, difficulty level of the course, and the expected grades from the first iterations of Versions A and B are discussed. There were 39 respondents out of a class of 76 for Version A and 57 respondents out of a class of 108 for Version B. Fig. 4 looks at the mean values of students' perception of their achievement of the Course Learning Objectives (CLOs) outlined in section II-B, on a four-point scale. The migration to a remote implementation has not affected the accomplishment of the CLOs. Zooming in on CLO3, students have felt that they are better able to plan the project in Version B versus Version A. This could be an added benefit of remote implementation and online learning as students have relatively more time at their disposal. A slightly higher percentage of students also feel that the team collaborations (CLO5) are better in Version B. This could be a result of not having to confront each other face-to-face which is usual in a project setting during a discussion resulting a better appreciation of their teammates.

Students perceive Version B to be of a higher difficulty level as compared to Version A, as shown in Fig. 5. This is expected due to the increase in credits, the corresponding redesign, and the COVID-19 remote implementation requirements. The surprising trend, though, is that the expected grades for Version B are, on average, better than those of Version A (Fig. 6). This is welcome feedback as it shows that students feel they are well-equipped to handle the higher difficulty level of the course.

The students were also asked the following two qualitative questions regarding the course:

Q1 What they liked about the course?
Q2 What they did not like about the course?

Some of the selected qualitative feedback from students (positive as well as negative) in the first run of Version B are given below.

- " We had the opportunity to work with people we really didn't know at all. A lot of experimenting and research is needed for every component. Actual hands on integration of different parts of the project which is valuable experience for our future"
- " Challenge the individual's ability to create a subsystem that is able to integrate with 5 other individuals. Very fun but challenging module that also tests the softskills that are required when working together in a professional environment."
- " I liked that we had an opportunity to do the project despite the pandemic. The adaptations also made it easier to work with the team as most sessions could be carried out online, from our respective homes at our convenient timings."
- " The fact that it is a project module but not FTF (face-to-face)"

- “ covid so meeting up is hard and integration is hard as well ”
- “ I would also suggest having a private, anonymous peer feedback session at the end of each group evaluation. This is necessary due to the hybrid nature of the project and other classes as teammates no longer see each other frequently.”

Overall, the comments focused on the general experience of the project and very few comments focused on the migration to the remote implementation. This was indicative of the fact that the remote implementation was smoothly integrated into the functioning of the project. Students experienced much excitement and a huge sense of achievement in this project in spite of the remote implementation.

B. Issues Encountered

- **Bluetooth Reliability**: BLE is inherently a reliable protocol for wireless communication, but its reliability is contingent on many extraneous factors. The leading cause of disruption is due to interference with WiFi as both BLE and WiFi share the 2.4GHz band. This issue was encountered when partial face-to-face classes resumed and multiple teams gathered in the same location with multiple BLE and WiFi devices powered on. Once the issue was discovered, the students were given guidelines on implementing resilient and reliable communication protocols in addition to the BLE protocols, and this was made a requirement for future iterations.
- **Firewall**: As with any organization, access to the network is largely at the mercy of the overbearing and restrictive network firewall. Over the years, the firewalls and the user authentication requirements have become progressively stringent. The access mechanism had to be modified multiple times to accommodate these changes. Initially, access to the network was as simple as accessing a common SSH server with a public IP address, which in turn was used to connect to the Ultra96. On this server, students were allowed to run their own scripts and programs, too. Later, this server was decommissioned and replaced with a dedicated SSH hop server, which required a preregistration of the public keys. Though these enhanced security features were uncomfortable, they did provide a great learning experience, with students exposed to real-world scenarios.
- **Ultra96 Physical Access**: A physical re-installation of the Linux system was required if a student deleted/corrupted any file on the Ultra96. This required the students to access the maker-space and physically swap out the memory cards of these devices. This was particularly difficult during the isolation restrictions, with strict permission protocols enforced for such access, and only the TAs were authorized to make these changes.
- **Ultra96 IP Address Allocation** : Each Ultra96 device is connected to the university network via a wired Ethernet interface. These devices must be registered with the network administrators to obtain an IP address from the Dynamic Host Configuration Protocol(DHCP) server. Each device also has a dedicated domain name. This allows students to access the devices without the knowledge of IP addresses, which are dynamically assigned. The issue with DHCP arises when a device reboots, causing DHCP to issue a new IP address. This IP address takes 4-12 hours to get updated on the local DNS server. As a result, the device becomes inaccessible using domain names. The only solution to this problem is physically accessing the device and noting the new IP address.
- **Hardware Sensor Units**: One key factor for the project's remote operation was ensuring each dancer had a functioning hardware sensor unit before the evaluations. This required planning to ensure the units could be passed to different dancers through common lockers or the TAs.

V. Conclusions

This study focuses on establishing a framework for designing a fully remote computer engineering capstone project. The COVID-19 pandemic resulted in the university migrating the teaching online for all courses. This was particularly challenging to implement for the capstone project as one of the key requirements is for the members to work together and subsystems to interact with each other. The theme of Industry 4.0 naturally fits the realisation, and the students are exposed to various I4.0 technologies as part of these sub-systems and the overall system. The different sub-systems are explored along with their interdependence and the usage of secure communication protocols to access them from the student's laptop remotely. The theme for the project used a cutting-edge problem: Human Activity Detection. The remote implementation of the project is integrated into the theme to ensure no disconnect between the different themes of I4.0 and Human Activity Detection.

The course was completely implemented remotely for its first iteration. There were teething troubles with collaboration amongst the teammates, but providing a comprehensive support system was monumental in finding solutions for the same. A summary of the issues encountered and the steps taken to address them are provided. As COVID-19 restrictions eased, the evaluations were progressively migrated face-to-face, but the project requirements requiring the use of I4.0 technologies were not changed. Understanding and implementing these technologies are key to exposing students to the current trends in the industry, and it also has the added benefit of shifting to a completely remote implementation if the need arises in the future.

References

[1] D. R. Herber, A. P. Deshmukh, M. E. Mitchell, and J. T. Allison, "Project-based curriculum for teaching analytical design to freshman engineering students via reconfigurable trebuchets," *Education Sciences*, vol. 6, no. 1, 2016. [Online]. Available: https://www.mdpi.com/2227-7102/6/1/7

[2] S. Cocco, "Student leadership development: the contribution of project-based learning, unpublished master's thesis." 2006.

[3] S. Wurdinger, J. Haar, R. Hugg, and J. Bezon, "A qualitative study using project-based learning in a mainstream middle school," *Improving Schools*, vol. 10, pp. 150–161, 07 2007.

[4] E. J. F. W. H. H. B. S. Bloom, M. Englehart and D. R. Krathwohl, *"Taxonomy of educational objectives: Handbook i"*. Cognitive domain. New York: David McKay, 1956.

[5] G. Mulholland and J. Turner, *Mulholland, G. and Turner, J.J. 2018. Enterprise Education in UK Higher Education: Challenges for Theory and Practice, 1-230, London: Routledge.*, 11 2018.

[6] H. Kondo and A. Hazeyama, "Systematic literature review on educational effectiveness of project-based learning for software development," in *2022 29th Asia-Pacific Software Engineering Conference (APSEC)*, 2022, pp. 584–585.

[7] L. D. Dunai, J. A. A. Daviu, L. Sava, and P. F. Dura, "Experiences on project based learning education," in *2023 IEEE 10th International Conference on E-Learning in Industrial Electronics (ICELIE)*, 2023, pp. 1–6.

[8] A. Bogdanovs, A. Bubovich, and I. Galkin, "Interdisciplinary project-based learning approach implementation for undergraduate electrical engineering students," in *2022 IEEE 63th International Scientific Conference on Power and Electrical Engineering of Riga Technical University (RTUCON)*, 2022, pp. 1–7.

[9] J. A. Marin, J. E. Armstrong Jr., and J. L. Kays, "Elements of an optimal capstone design experience," *Journal of Engineering Education*, vol. 88, no. 1, pp. 19–22, 1999.

[10] R. Bachnak, "An approach for successful capstone projects," in *Proceedings Frontiers in Education 35th Annual Conference*, 2005, pp. F4D–18.

[11] Accreditation Board for Engineering and Technology (ABET). Criteria for accrediting engineering programs, 2018 – 2019. [Online]. Available: https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2018-2019/

[12] R. M. Ford and W. C. Lasher, "Processes for ensuring quality capstone design projects," *34th Annual Frontiers in Education, 2004. FIE 2004.*, pp. S2G–12, 2004.

[13] W. Zhang, "Design of capstone projects under cdio mode," in *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, 2019, pp. 849–852.

[14] A. J. Dutson, R. H. Todd, S. P. Magleby, and C. D. Sorensen, "A review of literature on teaching engineering design through project-oriented capstone courses," *Journal of Engineering Education*, vol. 86, no. 1, pp. 17–28, 1997.

[15] K. Malur, Alumnus, P. Meena, and R. Deekshit, "A transformative approach to capstone projects in engineering," in *2014 IEEE International Conference on MOOC, Innovation and Technology in Education (MITE)*, 2014, pp. 348–351.

[16] L. Thames and D. Schaefer, "Software-defined cloud manufacturing for industry 4.0," *Procedia CIRP*, vol. 52, pp. 12–17, 2016, the Sixth International Conference on Changeable, Agile, Reconfigurable and Virtual Production (CARV2016). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827116307910

[17] B. Ritter, "Changing problem solving methods in higher education to meet the challenges of industry 4.0," in *2021 IEEE International Conference on Electro Information Technology (EIT)*, 2021, pp. 136–139.

[18] P. Caratozzolo, V. Lara-Prieto, C. Martinez-Leon, J. Rodríguez-Ruiz, R. Ponce, P. Vázquez-Villegas, and J. Membrillo-Hernández, "Developing skills for industry 4.0: Challenges and opportunities in engineering education," in *2022 IEEE Frontiers in Education Conference (FIE)*, 2022, pp. 1–5.

[19] I. B. Dhaou, "Aligning engineering education with industrial needs through specialized courses," in *2022 IEEE Global Engineering Education Conference (EDUCON)*, 2022, pp. 1664–1669.

[20] R. Fitoussi and H. Chassidim, "Teaching software engineering during covid-19 constraint or opportunity?" in *2021 IEEE Global Engineering Education Conference (EDUCON)*, 2021, pp. 1727–1731.

[21] A. Shekh-Abed and N. Barakat, "Transition of engineering education during the covid-19 pandemic," in *2021 World Engineering Education Forum/Global Engineering Deans Council (WEEF/GEDC)*, 2021, pp. 29–35.

[22] J. Svatos, J. Holub, J. Fischer, and J. Sobotka, "Online teaching of practical classes under the covid-19 restrictions," *Measurement: Sensors*, vol. 22, p. 100378, 05 2022.

[23] M. Mehrubeoglu, L. McLauchlan, D. Hicks, and A. Yeaman, "Engaged student learning through iot-based capstone projects: Particular look at student engagement in historically underrepresented groups," in *2023 IEEE Frontiers in Education Conference (FIE)*, 2023, pp. 01–07.

[24] D. A. Umphress, T. D. Hendrix, and J. H. Cross, "Software process in the classroom: the capstone project experience," *IEEE Software*, vol. 19, no. 5, pp. 78–81, Sep 2002.

[25] A. Cheville, "Designing successful design projects," in *2010 Annual Conference & Exposition*. Louisville, Kentucky: ASEE Conferences, June 2010.

[26] P. Brackin, D. Knudson, B. Nassersharif, and D. O'Bannon, "Pedagogical implications of project selection in capstone design courses," *International Journal of Engineering Education*, vol. Vol. 27, pp. 1–10, 01 2011.

[27] T. Wang, Y. Gan, S. D. Arena, L. T. Chitkushev, G. Zhang, and R. Rawassizadeh, "Advances for indoor fitness tracking, coaching, and motivation: A review of existing technological advances," *IEEE Systems, Man, and Cybernetics Magazine*, vol. 7, no. 1, pp. 4–14, 2021.

[28] S. Nooruddin, M. Islam, F. A. Sharna, H. Alhetari, and M. N. Kabir, "Sensor-based fall detection systems: a review," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, pp. 1–17, 04 2021.

[29] D. Arifoglu and A. Bouchachia, "Activity recognition and abnormal behaviour detection with recurrent neural networks," *Procedia Computer Science*, vol. 110, pp. 86–93, 2017, 14th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2017) / 12th International Conference on Future Networks and Communications (FNC 2017) / Affiliated Workshops. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050917313005