CS 4485 COMPUTER SCIENCE PROJECT

FINAL REPORT

UTD JSOM - Mobile App for Data Visualization and Analysis from Sensors

TEAM MEMBERS

Ameya Patil
Mason Orsak
Yash Bhalavat
Dhruv Malhotra
Ritika Villuri

ABSTRACT

The purpose of our project is to demonstrate our abilities in areas of Machine Learning and Mobile Application Development to create a power usage warning, risk mitigation application via the data gathered from sensors. The application would allow the users to understand, visualize, analyze and adapt their energy consumption habits and could aid in minimizing usage whenever possible. Additionally, the user involvement is minimal as machine learning models will generate warnings based on detected anomalies and provide appropriate feedback through notifications. This application aims to provide the foundational framework into integrating smart IoT devices and extracting meaningful information which can potentially reduce operational costs for various establishments.

Table of Contents

INTRODUCTION 3

DISCUSSION 3

RESOURCES 4

KEY ROLES 4

COMMUNICATION PLAN 5

RISK ANALYSIS/CONTINGENCY PLAN 5

COSTS 6

TIMETABLE (Weekly Schedule) 7

EVALUATION 8

CONTACT INFORMATION 9

IMPLEMENTATION DETAILS 10

CONCLUSION 10

SOURCES (REFERENCES, BIBLIOGRAPHY) 11

APPENDIX 11

IMPACT AND SECURITY 12

INDIVIDUAL ASSESSMENT 12

- Ameya Patil 12
- Dhruv Malhotra 12
- Mason Orsak 12
- Ritika Villuri 13
- Yash Bhalavat 13

ISSUES AND LESSONS LEARNT 14

FUTURE WORK 14

ETHICS DISCUSSION 14

Print Name/Signatures/Date: 16

I. INTRODUCTION

Electrical devices and appliances in our homes and various establishments that use too much energy contribute to both increased electricity costs and CO_2 emissions. People do not have numerous options to visualize their energy consumption habits and visualizing how and when they are consuming energy will aid in reducing consumption wherever possible. Our goal is to inform consumers of energy use anomalies, like a left-on stove or electrical short, so they can take steps to optimize their devices and reduce their overall energy costs and carbon footprint. Using our app the user could see what appliances are using the most electricity and begin to diagnose why by analyzing usage history and abnormalities. Once the problem has been identified steps can be taken to reduce energy consumption, such as repairing faulty equipment or buying more energy friendly appliances. The user can then ensure their implemented solution worked by comparing energy usage data from before and after their solution. Utilizing our ios phone application provides users the opportunity to check relevant data per appliance (like power usage, humidity, temperature, etc) in easy to digest graphs, which is not currently offered in the realm of energy consumption. In the status quo, energy providers offer mere data only about energy consumption, which doesn't offer descriptions for energy consumption per appliance. Understanding, analyzing, and visualizing each appliances' energy consumption gives valuable insights into energy-saving techniques resulting in reduced energy bills. Users can get updates, based on their usage-vs-weather relationship, and these updates will result in better energy management to reduce consumption. The usage-vs-weather relationship will be generated by the ML models to detect and warn the user of any forthcoming anomaly.

II. DISCUSSION

Our objective is to develop an app to display data received from sensors (such as Temperature, Vibration, Humidity, Air Pressure, etc) on a mobile device (e.g., iPhone, or iPad). The sensors should be purchased and installed by the users to measure their energy consumption along with weather data, the weather data can be gathered by location (see "Resources Needed to Use the Application" section for examples). We will develop predictive models using machine learning methods. The purpose of the models is to establish thresholds for energy consumption, should the consumption exceed the threshold the models will generate alarms when unusual conditions/events are detected. An unusual condition/event could be caused by a faulty application, or user-driven faults, such as leaving the fridge door open, not shutting off the stovetop, etc. These alarms will be sent to the mobile device so that the user becomes aware and makes appropriate decisions/takes suitable actions in order to minimize and prevent over usage. A cloud-based solution should be implemented with an intuitive UI. AWS integration will allow

for more users to connect to this service without the need to expend their own devices' storage and/or computation power.

III. RESOURCES

Our team consists of 5 UTD students, faculty advisor, and a technology advisor. All of our team-members will be utilizing macOS as our preferred OS choice.

Environments, tools, frameworks, programming languages and software packages to be used:

- Mac OS
- XCode
- VS Code (2)
- SwiftUI
- GitHub
- Jira
- Lucid Chart (for UML creation)
- Google Suite (Docs, Sheets, Colab)
- SQL
- MySQL Workbench
- AWS Relational database
- AWS Sagemaker

IV. KEY ROLES

- **Ameya Patil**
  - Skills: Python, ML, C++, Java
  - Tools/Env: VSCode2, AWS, GitHub, XCode, macOS
  - Domain for Project: Back-End, ML
  - Role: Team Lead, POC for Mentor
- **Mason Orsak**
  - Skills: Python, SwiftUI, C++, Java, SQL
  - Tools/Env: XCode, VSCode, GitHub, Windows 10, macOS
  - Domain for Project: Front-End, SwiftUI
  - Role: Track of Meeting Minutes
- **Yash Bhalavat**
  - Skills: ML, Java, JS, SwiftUI, MySQL
  - Tools/Env: JMP, Pycharm, AWS, GitHub, Xcode.

    - Domain for Project: Back-End, ML
- **Dhruv Malhotra**
    - Skills: Python, SwiftUI, SQL
    - Tools/Env: Sequel Pro, Xcode, AWS, macOS
    - Domain for Project: Back-End, Databases
    - Role: Quality Assurance/Tester, Comic relief
- **Ritika Villuri**
    - Skills: Python, SwiftUI, Java, SQL
    - Tools/Env: Xcode, GitHub, PyCharm, MySQLWorkbench, macOS, Windows 10
    - Domain for Project: Front-End, SwiftUI

V. COMMUNICATION PLAN

Communication is significantly important to make progress given the scope of the project. We broke down the project into 3 parts/aspects: machine learning, front-end (UI), and back-end (databases). We meet with our mentor and technical advisor every Tuesday to update them on the weekly progress and get input regarding the next phases and steps we need to take. We also converse, via email or Teams, with our technical advisor should we need assistance on our project. Additionally, we meet in small groups (2-3) members, based on the aspect we are working on on a daily basis. All group members meet on Friday for updates, reports, and concerns. These aforementioned member meetings take place via voice-calls on Discord.

VI. RISK ANALYSIS/CONTINGENCY PLAN

Potential Risk	**Likelihood**	**Impact**	**Risk Mitigation Procedure**
Power outage/Loss of Internet due to inclimate weather	Medium	Low-High. Internet failure **usually** resolves itself in a relatively short time, but extended power or internet outage might lead to team members being unavailable for long periods of time	If the issue stems from a natural disaster or any sort of non-local event, then the plans can be delayed until the internet/power can be back online. If it is a local hardware issue then remaining work will be split up among the other team members.

AWS Functionality Loss	Low	Would hinder the data storage, inbound and outbound, and would paralyze the machine learning capabilities	If the issue is a local problem specific to the AWS account, team members will use other members' accounts as a redundancy.
Team members contracting an illness and as a result are not able to work	Medium	Medium-High depending on the illness.	Work from that team member will be divided among the other team members
Two or more members involved in a personal disagreement	Low	Low. Team is composed of professionals and disagreements can be worked out quickly.	Members involved in the disagreement will discuss with the larger group. Will have a vote about the issue to resolve it. If the issue is unresolvable, supervisors might be included.
Team member is not carrying their weight in terms of project completion	Low	High. This will cause problems within the team and will delay production of the project as a whole	Members who notice this will notify team leadership and handle it internally. If it is unresolveable, we will bring supervisors to intervene.

VII. COSTS

Item	Description	Cost
Apple Developer Program	Allows submission of applications to the AppStore for deployment to the user.	$99 per membership year
Raspberry Pi 4	Receives data from each device sensor and inserts the data into the AWS database.	$0, provided by faculty
Raspberry Pi Sensors	Sensors that allow measurement of energy usage, temperature, humidity, etc.for each device.	$0, provided by faculty

AWS Credit	Credit for any AWS services needed for the project, most of the credits went towards the database.	$50 per group member

FINAL COSTS:

$0, as the project was developed on Free-Tier using AWS Educate. We did not purchase the aforementioned sensors and electronics, the Apple Developer License was not purchased either, development was done on local Macs and the code is hosted on GitHub for free.

VIII. TIMETABLE (Weekly Sprints)

Start Date	End Date	Task
1/25/21	1/31/21	● Planning and diving initial work based on meeting. ● Setting up technologies like Github, Jira, Colab and more. ● Figure out if any additional tools and costs are associated. ● Learning Swift UI, ML techniques for analysis.
2/1/21	2/7/21	● Re-preprocessed the dataset to reflect the changes on our new problem statement ● Created a group-specific Github account to upload code, so all team members have access. ● Created markup of front end UI using marvelapp.com. ● Continued Machine Learning Course and Swift UI course.
2/8/21	*2/14/21*	● Researched anomaly detection packages for ML. ● Create UML Diagram of overall app infrastructure ● Task Tracker and GANTT chart created and populated.

		• SWIFT UI : Landing page and buttons created
2/15/21	*2/21/21*	• Started Project Proposal • Implement anomaly detection packages and perform additional feature-engineering wherever necessary. • Create framework for UI detail views data • A big impact of power failure as most team members did not have power.
2/22/21	*2/28/21*	• ML: Completed Feature Engineering to determine which features have significant correlation to appliance energy consumption. • ML: Learning Anomaly detection techniques and started to implement anomaly detection using Isolation Forest. • Begin implementing notifications in UI
3/1/21	*3/7/21*	• AWS RDS database setup and wrote python scripts to add it. • Find the major Anomaly in the dependent variable of the dataset • Implemented AWS database architecture and inserted subset of kaggle database • Implemented database into UI • Configured Raspberry Pi and connected it to AWS database. • Have Project Proposal ready.
3/8/21	*3/14/21*	• Setup sensors on Raspberry Pi to transmit data to AWS and pull results into UI • Fully implement ML functionality on AWS • Have the models ready and working on adding it to the architecture.
3/15/21	*3/21/21*	• Create more displays in UI data views for easy viewing. • Merge codes based on the architecture we designed.

		● Application testing and have the first demo ready and present it.
3/22/21	*3/28/21*	● Take feedback and apply it onto our application. ● Do more testing and add functionality.
3/29/21	*4/25/21 (4 weeks)*	● Still continue application testing and development and have the final product ready by this period.

IX. EVALUATION

As a team, with feedback from our mentors we have come to the conclusion that our main measurement for the success of the project would be a fully functional intuitive and user-friendly iOS mobile application that will hold all of our data without any interruptions. To ensure the success of our project we have chosen to implement an Agile methodology with JIRA to maximize efficiency across our team. Given that our teams have met and have been functioning fully remote it was important to choose this methodology.

Listed below are some performance metrics that we measured to gage how well our project was built.

- Average of 1.27 second API call response time querying 500,000 database entries
- Relational database read and write latency average 0.0004 seconds
- Run more resources intensive operations at regularly scheduled intervals using AWS EventBridge to avoid long API wait times
- IOS application loads only the data the user is currently requesting to further increase UI response times
- API calls are performed asynchronously to application use, avoiding app freezing during data requests
- Entire AWS backend implemented solely using AWS free tier
- Our ML model current performance score is at 2.17 %

X. CONTACT INFORMATION

- **Ameya Patil**
  - Contact Information:
    - Phone: 469-512-4666
    - Email: arp160730@utdallas.edu
  - Short Bio
    - CS Undergrad, System Eng. & Mgmt. (Fast-Track, Grad)
    - Graduating: Spring 2021
    - Interests: Investing (stocks, options), music, reading, watches and cars.
- **Mason Orsak**
  - Contact Information:
    - Phone: 702-756-8650
    - Email: mto180000@utdallas.edu
  - Short Bio
    - CS Undergrad, System Eng. & Mgmt. (Fast-Track, Grad)
    - Graduating: Spring 2021
    - Interests: Music, science fiction, video games.
- **Yash Bhalavat**
  - Contact Information:
    - Phone: 972-802-7125
    - Email: ykb180000@utdallas.edu
  - Short Bio
    - Graduating: May 2021 (CS undergrad)
    - Interests: Coding, Music, Movies, Learning new technologies.
- **Dhruv Malhotra**
  - Contact Information:
    - Phone: 8329181419
    - Email: dxm180013@utdallas.edu
  - Short Bio
    - Graduating: December 2021 (CS undergrad)
    - Interests: Performing Stand-Up Comedy, blockchain development.
- **Ritika Villuri**
  - Contact Information:
    - Phone: 469-912-5442
    - Email: rxv163430@utdallas.edu
  - Short Bio
    - CS undergrad
    - Graduating: December 2021
    - Interests: Hiking, TV shows, Fashion, Badminton, American Football, Cricket.

XI. IMPLEMENTATION DETAILS

Just in case our project was to be picked up and continued by another team we spent about two weeks at the end of the semester gathering quite extensive documentation about our implementation for every part of this project. It contains around 13 pages so we decided to leave it as a separate report and attach it along with this document for our submission, please refer to it for specific implementation details.

XII. CONCLUSION

Despite this project not reaching a level of completion that we originally hoped to see at the beginning of the semester, we believe it to still have significant utility. The work we accomplished not only provides a proof of concept for data visualization using real time sensors with a machine learning anomaly prediction, but also set a solid foundation in which another senior design group could build off of to achieve the project's original goals and much more. While the project is not ready to be used in a lab environment as the project mentors hoped, It has the functionality and architecture where just a few more weeks of development would allow a switch to live data pulled from sensors and significant improvement in anomaly prediction accuracy. By the end of this project our development team had acquired many new skills like project management, front end development, database administration, AWS backend development, machine learning, and much more. We would like to thank Dr. Shrivastava and Mr. Mishra for their much needed guidance throughout this project as well as Dr. Razo for organizing this course. Please feel free to direct questions about this project to any of its team members whose contact information can be found under section 10 “Contact Information”.

XIII. SOURCES (REFERENCES, BIBLIOGRAPHY)

1. Source Code
    - https://github.com/masonorsak/CS4485.001
2. Original Dataset
    - https://www.kaggle.com/taranvee/smart-home-dataset-with-weather-information
3. AWS Database Import
    - https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/MySQL.Procedural.Importing.NonRDSRepl.html
4. Data Analysis (Python)
    - https://colab.research.google.com/drive/1IZNrpIAGgReQQ4c7lW5kGAOhwKwfql2i?usp=sharing
5. API Gateway Creation
    - https://medium.com/@hk_it_er/create-lambda-and-api-gateway-nodejs-aws-serverless-to-rds-mysql-6a75243e61cc

6. SwiftUI - The Complete Developer Course and SwiftUI Bible
    - https://learning.oreilly.com/videos/swiftui-the/9781801070676/
7. User Interface GitHub
    - https://github.com/masonorsak/SeniorProject
8. Graphing in SwiftUI
    - https://www.youtube.com/watch?v=ZtOMxubWklw

XIV. APPENDIX

- Project documentation was submitted along with this document and outlines in detail our implementation of the project goals
- AWS Documentation
    - https://docs.aws.amazon.com/
- Node.JS Documentation
    - https://nodejs.org/en/docs/
- Python 3 Documentation
    - https://docs.python.org/3/

XV. IMPACT AND SECURITY (Finish-up, Ameya)

The impact of our project is to allow users to learn about their energy consumption habits and to detect anomalous behavior when it occurs so that the user can make necessary changes in their energy consumption. The way they can learn about their energy consumption is by tracking the energy by each of the connected appliances. The data collected will provide insights into the time at which the device was active, the energy consumption, the corresponding weather attributes at the given, specified time. These features are then utilized by the machine learning algorithm to track the correlation between the weather features and the energy consumption by the respective appliance. Once these relationships are mapped and calculated, the user can be warned and alerted should an anomaly be predicted by the ML model. Although the ML algorithm is running in the back-end the user can view the necessary statistics and alerts on his/her iOS device and share them with peers or other relevant parties. On the security aspect, we're using a publicly available data set as listed on Kaggle (refer to references). For that reason, we did not implement a layer of security on the data side, however, on the UI front we allowed user profile creations such that the data could be displayed on the device if the user has logged in with the appropriate credentials.

XVI. INDIVIDUAL ASSESSMENT

Ameya Patil

For this project, my contribution was for machine learning and AWS. At the beginning of this semester, we all decided to equally contribute to the user interface as that was meant to be the most time consuming. Since we all were new to front-end development, we decided to learn SwiftUI as the choice of device implementation was going to be on iOS devices. Once the buttons and flow diagrams were created, I decided to work on machine learning and get a model implemented so that it could detect anomalies and be ready for the front end. Firstly, we started with preliminary data analysis for the given data set in order to draw a better understanding of the data, so that we could develop appropriate models. The way I conducted data analysis was with use of built in Python libraries and built-in statistics models. Our goal was to identify any correlation between the features listed in the data set. We hoped to find correlation between the weather data and energy consumption for the appliance, but unfortunately, there was little to no correlation between those features. Hence, based on the guidance of our technical advisor (Mr. Sailendra Mishra) we decided to implement a linear regression model which would use time as the independent variable to predict consumption at time X in the future. The functionality of this model was to act as a placeholder in our AWS SageMaker service as the model can be easily swapped out with a better, more accurate model. Before getting to the stage, Yash and I implemented other models within our domain knowledge. These models include k-NN, neural networks, and a long short term memory model. Once we found out that these models did not produce accurate results, we decided to refer to online resources in order to research a better methodology while dealing with a time series data set. We found that LSTM or an SVM model would be beneficial towards increasing the accuracy. Due to our limited knowledge regarding implementations of the aforementioned models, we decided to shift our focus towards getting the AWS architecture in place and functional. I collaborated with the deployment of the Python model in the ML service on AWS. Once the model was deployed and functional, we then had to populate the relational database that stored the data and interacted with the device using SQL queries. Once the anomaly column is populated the device can then display the appropriate graphs and statistics for the device. Getting the data from RDS to the device was challenging as we had to connect to an API gateway which would allow communication between the device and the RDS. The link was established using Lambda functions EC2 instances which would allow the user to retrieve necessary information to populate the graphs and tables. For the task of populating graphs, there were a large number of data points and the functions used to populate the graphs were too big in size to be supported by the Lambda functions. In order to remedy this issue, we decided to utilize an EC2 instance which would give us more access to computing power and support larger data files. Lastly, we focused our efforts to make the graphs somewhat interactive and informational so the user could visualize the data points on the graphs.

Unfortunately, we were unable to do that as the necessary libraries had poor documentation and the data being populated was too large to scale down.

Aside from the technical contributions, as the team leader I was responsible for creating agendas, establishing deadlines using Gantt charts, adhering to any concerns our group members had and assigning tasks based on the liking and interest of our team members. Additionally, I created flow diagrams in order to supplement the code documentation which would allow the user to track the workflow and the data flow in the back end and front end for our application. Overall, this was a great learning experience as I learned a great deal about full stack development, AWS services, and client interaction. It was a great opportunity to apply the concepts I learned while acquiring the AWS cloud practitioner certificate. I'm looking forward to applying these new concepts I learned during my career as a graduate student and beyond when I enter the tech industry. The lessons and knowledge acquired through application of knowledge in our preexisting confines and adding to our skillset was an invaluable experience. This project offered a glimpse into the real-world projects which could have a drastic impact on society and culture and changed my perspective towards web and app development.

Dhruv Malhotra

I chose to focus my time on assisting with the database, implementing testing infrastructure, and assisting with the iOS development, specifically allowing for sharing information through social media sites. I am currently working at a software engineering startup and as a result I had access to various developer tools to assist in our overall development of the app. When we first started this project we tried to use a local server to store our data with the plan of being able to access it remotely but we ran into logistical issues with problems being able to access the data. Unfortunately, we ran into connection issues, lag times, and storage issues so we had to switch gears. We decided after that to pivot to using Amazon Web Services to store our kaggle dataset and after a bit of trial and error, managed to get it up and running.

On the testing side of things I helped set up our testing infrastructure to facilitate the movement of tickets to move sure we added features on a regular basis. I have some experience with Quality Assurance and as a result I felt it was important to give our user perspective when creating the application. Since the goal of this project was to be able to upload this on the App Store so I felt it was important to enhance overall user experience. If there was a bug in the UI I would bring it up to the team and we would go about fixing it and then adding the github branch to correct what needed to be fixed. Normally it would include decreasing load times for data to be presented on screen.

In addition to that I took part in creating the settings page so that whenever retrieved the data from a particular appliance energy we could personalize it to an individual MAC address. The various logins we used were Login via Facebook, Twitter, or Google. This would allow the user to create an account and have the information adjusted for their individual energy usage

information. Since most people have a login for at least one of the organizations mentioned earlier this would allow us to glean more data from the overall user. We also wanted to be able to adjust notifications on the phone so that if there was an anomaly, we could notify the user that a particular appliance was using too much energy and inform them accordingly. We allowed sharing functionality by importing a Facebook API so that the user could show what their user data was with other people who were on the app. Finally we also have a mailto API that allows for the user to reach out to us directly if they had any technical issues as well as rate our app on the App Store. These were all necessary so we could get direct feedback from our users and adjust accordingly to improve the overall experience.

I've learned quite a bit from this project. On the technical side I've learned machine learning algorithms and how they can be implemented, I've learned how to visualize the data so it can add value to a team, and how data can sometimes cause more problems than they can solve. I've also learned about project planning, including that not everything works as easily as you want it to, and if there are issues it's important to switch gears and act accordingly. I have also increased my overall confidence in becoming a software engineer as I have transitioned into a full time role at my current organization.

Mason Orsak

I chose to focus my time for this project on development of the user interface, database, and AWS architecture since I am the only person in the group who was not currently enrolled in or had already completed a machine learning course. I began the semester by completing a O'Reilly course (reference 6) on SwiftUI since it is a new language for me. This is also where we got the idea for how to structure the foundation of our application (using an app data file to call the api, see project documentation). Next I began developing the database architecture by modeling it with a ER diagram in MySQLWorkbench. Once we had this architecture approved by the project's technical mentor (Sailendra Mishra) I used MySQLWorkbench's forward engineering feature to get a sql script from this ER diagram. From here, with the help of my teammates, we developed a python script that would import data from the kaggle dataset (reference 2), load it into a pandas dataframe, then used pymysql to connect to a local instance of the database and commit all the data to the architecture we created. Once we confirmed this data was correctly imported to the database I created a AWS relational database (RDS) and used this same script to import a subset of our data which would allow us to continue development of the AWS architecture. From here I spent most of the remainder of the semester creating lambda functions that were triggered by API calls to AWS API Gateway which queried the database, performed some manipulation of the data, and sent that data to the caller of the api. I also spent the remainder of the semester developing the user interface, its communication with the AWS backend, and displaying our data in graphs.

I learned a lot of new technologies for this project. AWS, SwiftUI, and machine learning were all new to me and I had to learn on the go. Despite the many developmental roadblocks we encountered during the development of this project, I enjoyed this project overall and learned a

ton along the way. I now feel much more familiar with full stack development and look forward to using my new skills in pursuit of my academic and professional goals.

Ritika Villuri

For this project my contribution was mainly towards the front-end development and UI of our mobile application. Since this was our first time working with Swift, my teammates and I spent the first few weeks learning and familiarizing ourselves with Swift Programming and all the various features and possibilities within it. I kept learning as we were working on the UI as well. Within the first few weeks of learning, we made the homepage of our application and added simple buttons and pages and tested it out. We did not have any real data to connect to it yet so the main focus was just to get the application up and running to make sure that it looked nice and that it was user friendly. The application is what the audience was going to look at and that is the by-product of our entire work, so we had to make sure that we made it look as best as we could. All of this was possible with the help of many courses available on the internet for SwiftUI. One of the courses that my teammates and I mainly referred to was the O'reilly course for Swift.

The next focus was getting our database populated with the dataset we were working on. We started off by making ER diagrams first to really understand how we were going to structure our architecture. My teammates developed a python script that would pull the data from the kaggle dataset. Most of the database and machine learning (back-end) part of the project was taken care of by some of my teammates that had previously taken the course in ML and had experience with it. We moved all our architecture to the free tier in AWS and had connected that to our mobile application.

The remaining part of our project, I spent a lot of time trying to make interactive graphs that would truly visualize all the data we analyzed and provide the users with a good view of their energy consumption. It was difficult to navigate around and find the right resources for the development of these graphs in the UI as this was my first time learning the language and I could not find many resources from Apple directly on creating these graphs. I did some research and found some packages people posted on GitHub that assisted with creating graphs. I tried to implement that within our code and we faced roadblocks when it came to customizing it our own way, with labels and legends since it was a package made by a third party. I reached out to the person that uploaded the package on creating these graphs and tried to make it work.

Overall, this project has been a huge learning curve for me. I have learned so many different things, like Swift coding, understanding the value of pre-processing data, making training models, and the importance of proper project management. The project also gave me a great bird's eye view of how a software development cycle would actually work in the real world.

Yash Bhalavat

During this project I worked mostly on the ML, AWS and also a bit on the UI side . I started this project by trying to build an effective model for our data that we collected from the Kaggle dataset. The dataset was a time-series dataset which means that there is new data coming in real-time and so we need to keep time as an independent factor in our model. Since this was the first time working with time series data I started learning about different models like LSTM, CNN, ARIMA using o'Reilly course and Youtube. After that me and Ameya started working on the dataset and built some static models like Neural-nets and K-NN to find correlation but we didn't find correlation between any columns and so we started with building a LSTM time series model. After starting we were having a lot of issues with the model like load time issues with epoch, not accurate predictions and so resolving that took a lot of my time. After that all the team members and our technical mentor (Dr. Sailendra) decided that we should just have a base linear regression model and work on connecting it to the UI and then return again to replace the model. So after that I built the Linear regression model with the help of our technical mentor and then I put the model on AWS sagemaker which we use to build the model and then I stored the model into a pickle file in an S3 bucket. So after the model was loaded I started working on the AWS lambda functions along with Mason to have our model connect to our database and then we wrote some queries which would basically get the data from the database, run the model on them and then update the anomaly column in our database to 1 but we were having some issues with loading such large dataset on lambda and so we pivot to EC2 instance and then it worked fine. After that during the end of the project I started learning Swift UI and then started working on ways to make our current graph look better. After working I figured that the issue is the large points we plot and so we decided to have a new table where we would get the average consumption of each day and plot that but due to time constraints we weren't able to finish that part. So that was my overall role on the project. Although we haven't achieved our goals, I put in a lot of effort and also got a chance to learn a lot of new skills. I believe this project will help me a lot in my professional career. I hope I can continue to work on this project in the future personally if our mentor permits.

XVII. ISSUES AND LESSONS LEARNT

Ameya Patil

Initially, the biggest issue we faced was determining how to implement the front end and back end to achieve the desired goal. After a couple of days of discussions, we settled on developing the front-end using SwiftUI in the back end using Python. The reason I chose Python for the back end was based on my previous experience in my machine learning class. Additionally, Python offered a wide variety of libraries and built-in functions which would allow us to implement models with relative ease. My hurdle for the front end was that I had no knowledge of using SwiftUI for IOS application development. The next hurdle I faced was

deciphering the relationships between the features in our given data set. With the aid of Python libraries, I was able to generate graphs and histograms to spot trends and using heat Maps see correlation between the features that were present in the data set. Our mentor (Dr. Prakash) gave us access to the JMP Pro 15 tool which made data visualization and statistical analysis a little easier. Once we found that there was little to no correlation between the desired features, we had to implement a placeholder model and sacrifice accuracy in order to generate a model that would populate the respective columns in AWS service. We developed the model on our local computers and then decided to migrate to AWS SageMaker. The problem we faced there was we were training the model every time the sage maker instance ran and because our data set is static and not dynamic, we were wasting computing power and increasing operational costs for our machine learning back end. In order to remedy this issue, we decided to convert the model to a binary file after being trained on a local machine and supplying the binary file to the AWS SageMaker such that the model would run on the analysis and anomalies population part and need not be trained repeatedly with each instance. The next hurdle was to populate the respective RDS column such that the anomalies would display and populate the graph on the front end. Once the RDS columns were populated we had to generate SQL queries that would pull the relevant data to plot and display on the application. The connection between the device and the RDS database was challenging however we were able to connect the two using the AWS API gateway and Lambda functions to pass the SQL queries and return results which would then be fed to the device for population and display purposes. Our final challenge was to make the UI as user-friendly as possible and adding security features such that user profiles could be generated and the data would be limited to the user itself. Due to the shortage on time, we were not able to fully implement our desired goal of displaying the graphs in a neat tidy fashion, however, we're able to display the relevant and important information that the user might need in order to track where the anomalies are occurring and the averages such that they provide insight into energy consumption of the appliance.

We were able to overcome these issues with weekly discussions with our technical advisor (Mr. Sailendra Mishra) who provided us with timely insight and gave us the necessary direction in order to implement the functionality in the most efficient way possible.

Dhruv Malhotra

When we first started working on the project, we had to spend a lot of time learning new technologies. If we hadn't learned about a particular functionality, we had to learn it on the fly. For example, I was in the process of taking a machine learning course and as a result wasn't fully versed in the subject matter when the project started. When we first started this project, we tried to use a local server to store our data with the plan of being able to access it remotely, but we ran into logistical issues with problems being able to access the data. Unfortunately, we ran into connection issues, lag times, and storage issues so we had to switch gears. We decided after that to pivot to using Amazon Web Services to store our Kaggle dataset and after a bit of trial and error, managed to get it up and running. On my end, when implementing iOS development, we had some issues porting Facebook APIs onto our app itself. The idea we had was to login via your Facebook account as an ability to personalize the data. Unfortunately, because we had not set up an Apple Developer License, this caused some issues when development which wouldn't port properly. We found a workaround, but it doesn't have the same functionality as the login credentials that are used by other apps. Google and Twitter Logins were much easier, but since most users have a Facebook login it will not have the same user profile logins data aggregation techniques.

One thing I've learned through the project is that not everything that works on paper will work in practice. I also learned that we have to learn things on the fly when it comes to creating a new product as opposed to developing something that has been in the works for quite some time.

Mason Orsak

One of the largest issues I had to overcome during this project was lack of experience. Just about every service, programming language, and technology was something that I had either very little experience in or none at all. This was my primary concern going into this project which is why I spent the first few weeks of the semester trying to learn as much as I could. This began with completing an O'Reilly course in SwiftUI (reference 6). I then began working on freshening my python and node.JS skills since it became apparent they would make up the bulk of code for backend development. I achieved this by simply reviewing old notes and powerpoints I had accumulated through my years at UTD. Finally, I tried to get the basics of AWS inter service communication down by reviewing their documentation. This was greatly helpful as it helped me later diagnose many of the problems we would encounter with virtual private cloud and security groups. Once my lack of experience in these areas was addressed I then turned my attention towards my second greatest concern, deadlines. As anyone who has a background in computer science can attest to, predicting when a programming assignment will be completed is difficult. As anticipated we would regularly run into hiccups in our development as unforeseen complications arose. The accumulation of these delays in development is what ended up pushing a project back and ultimately ended in us not being able to complete all the features we would have liked to. Hindsight is 20/20 but if I had to do this project over again having better

communication with my team and clearer distribution of tasks that need to be completed would be my focus. I would begin addressing this by keeping up with our Jira management so each team member could see what everyone was working on and what needed to be worked on next. I would also try and schedule more regular team meetings to address any concerns that came up during development. Despite some of these shortcomings I am proud of what we were able to accomplish this semester and look forward to utilizing the knowledge acquired during this project in my future endeavors.

Ritika Villuri

This project has been a huge learning curve for me. One of the biggest problems I would have to say I had was that I had no working or academic experience of the tools we were using for the project. I had just enrolled myself in the Machine Learning class at the start of the Spring semester so I was just learning everything as we were working on the project. Having some prior experience in SwiftUI development would have really helped me and made things a little bit easier but that is what this project taught me. It taught me that we are not always going to be facing things that we already know or are really good at and being flexible and adapting to the fast-paced world in CS is what is going to work. Which is exactly what I tried to do while I was working on the project. I spent a lot of time reading and watching tutorials mainly on iOS mobile application development. I made mistakes and learnt to correct them as I was working on it.

Another issue that we did have was time management. It really taught me the importance of having proper and realistic goals set. Although we did have weekly agendas and meetings, I think utilizing project management tools such a JIRA and many more out there properly would have helped us truly understand what we were behind on and what we spent a lot of time and what we made significant progress on. Although we did utilize this resource and make Gantt charts to track our progress, there is always room for improvement. I have learned a lot from my teammates and my mentors (Dr. Prakash and Mr. Sailendra). Their constructive feedback throughout the semester really helped us grow and push ourselves to do better. I am grateful for the experience I got from working with the team and the mentors. I got the opportunity to learn tools such Swift and python and implement them in a live working project.

Yash Bhalavat

As we know every project comes with its own issues and during this project I did face some issues the one that took the most time was having an accurate model for our dataset. Since I have never worked with time series models and as the time was also in Unix time it took me a bit to understand how to work on those types of dataset. I spend a lot of my time learning about how time-series prediction and forecasting happens using LSTM, CNN and ARIMA. Later once I started using LSTM I faced a lot of issues like a long epoch time. It would take about 25 min sometimes for it to have 30 epochs and still not be accurate and so we and our mentor decided to move to the next step of connecting ML to UI. Another issue I faced was with Lambda functions

where it was a size barrier and it took us about 1-2 weeks of trying different things like changing queries, trying different libraries, using AWS cloud 9 and more but we finally used EC2 as an alternative and it worked. I also got a chance to work a bit on the UI during the last couple of weeks and so I started learning about Swift UI and so I was working on making the graph look nice and so I saw that the option is to reduce the points and plot average of each day to have a good scope for a month but to do that we had to add a new table in SQL, create python script to call and more and so I decide to do it manually for some graphs which did look nice but the time constraint stopped us from completing it. Even though I faced a lot of issues, this project provided me an opportunity to learn a lot of new skills like building time-series models, AWS sagemaker, AWS lambda, Swift UI and also our mentor (Dr. Prakash) teached a lot about team management and planning. Also with learning it also gave me experience of working in a team environment and Agile methodology. It also helped me understand how sudden things change in professional life. I was planning on doing more work but due to time constraints I wasn't able to fully finish the project but I hope our work would help the next team get a good start and can finish it. I lastly want to thank both of our mentors for helping us throughout. They were very much invested in this project and helped us a lot throughout the way.

XVIII. FUTURE WORK

Due to the limited amount of time we weren't able to add all the features we were planning on this project but we made our code such that it would make it easy for the next team to add those features. Some of the features we were planning are:

- Being able to read live data from sensors.
- Have the UI graph look more nice by reducing the points of the dataset.
- Have the functionality in UI that would allow the user to change the dates so they could see usage for those dates.
- Have a better ML model that works well with time-series dataset.
- Get the apple developer license to publish the app
- Change UI views from absolute positioning of UI elements to relative positioning to support phones with different sized resolutions
- Increase data security by having non public AWS database and s3 buckets

XIX. ETHICS DISCUSSION

Although we used datasets, as cited in references, we did not use or replicate code associated with it. Our objective was to integrate AWS services for machine learning and data storage and the UI was developed from scratch using SwiftUI. Built-in libraries and frameworks were used in development of the code, however, they are documented in the language/IDE documentation and are open-sourced. We did not face any ethical concerns regarding aspects of our code as the application's objective is to allow users to track their appliance's energy consumption. The energy consumption data is secured on the device using user profiles and the host (owning device) can choose to share the data with other users should they choose to do so.

XX. Print Name/Signatures/Date:

Ameya Patil Team Member 1 Ameya Patil Sign Here	*Mason Orsak* Team Member 2 Mason Orsak Sign Here
Yash Bhalavat Team Member 3 Yash Bhalavat Sign Here	*Dhruv Malhotra* Team Member 4 Dhruv Malhotra Sign Here
Ritika Villuri Team Member 5 Ritika Villuri Sign Here	
Technical Advisor: Sailendra Mishra Sign Here	Faculty Advisor: Dr. Prakash Shrivastava Sign Here

Signature: Dhruv Malhotra
Dhruv Malhotra (May 13, 2021 16:58 CDT)
Email: dhruvmalhotra2021@gmail.com

Signature: Mason Orsak
Mason Orsak (May 13, 2021 17:05 CDT)
Email: masonorsak62@gmail.com

Signature: Ritika Villuri
Ritika Villuri (May 13, 2021 17:06 CDT)
Email: rituvilluri1@gmail.com

Signature: Ameya Patil
Ameya Patil (May 13, 2021 17:07 CDT)
Email: aameya.patil98@gmail.com

Signature: Yash.k.Bhalavat
Yash.k.Bhalavat (May 13, 2021 17:09 CDT)
Email: yashbhalavat98@gmail.com

Final Report - Senior Design

Final Audit Report 2021-05-13

Created:	2021-05-13
By:	Sam Forever (sam.lov333@gmail.com)
Status:	Signed
Transaction ID:	CBJCHBCAABAA7-ega84gTFaiCwoTnaFMEVkF_uh2qiJA

"Final Report - Senior Design" History

Document created by Sam Forever (sam.lov333@gmail.com)
2021-05-13 - 9:54:50 PM GMT- IP address: 70.121.42.46

Document emailed to Dhruv Malhotra (dhruvmalhotra2021@gmail.com) for signature
2021-05-13 - 9:55:36 PM GMT

Email viewed by Dhruv Malhotra (dhruvmalhotra2021@gmail.com)
2021-05-13 - 9:55:46 PM GMT- IP address: 66.249.80.199

Document e-signed by Dhruv Malhotra (dhruvmalhotra2021@gmail.com)
Signature Date: 2021-05-13 - 9:58:34 PM GMT - Time Source: server- IP address: 76.187.194.49

Document emailed to Mason Orsak (masonorsak62@gmail.com) for signature
2021-05-13 - 9:58:36 PM GMT

Email viewed by Mason Orsak (masonorsak62@gmail.com)
2021-05-13 - 10:04:20 PM GMT- IP address: 99.99.196.7

Document e-signed by Mason Orsak (masonorsak62@gmail.com)
Signature Date: 2021-05-13 - 10:05:11 PM GMT - Time Source: server- IP address: 99.99.196.7

Document emailed to Ritika Villuri (rituvilluri1@gmail.com) for signature
2021-05-13 - 10:05:13 PM GMT

Email viewed by Ritika Villuri (rituvilluri1@gmail.com)
2021-05-13 - 10:05:24 PM GMT- IP address: 99.48.0.48

Document e-signed by Ritika Villuri (rituvilluri1@gmail.com)
Signature Date: 2021-05-13 - 10:06:35 PM GMT - Time Source: server- IP address: 99.48.0.48

Document emailed to Ameya Patil (aameya.patil98@gmail.com) for signature
2021-05-13 - 10:06:37 PM GMT

Email viewed by Ameya Patil (aameya.patil98@gmail.com)
2021-05-13 - 10:06:46 PM GMT- IP address: 64.233.172.103

Document e-signed by Ameya Patil (aameya.patil98@gmail.com)
Signature Date: 2021-05-13 - 10:07:25 PM GMT - Time Source: server- IP address: 70.121.42.46

Document emailed to Yash.k.Bhalavat (yashbhalavat98@gmail.com) for signature
2021-05-13 - 10:07:27 PM GMT

Email viewed by Yash.k.Bhalavat (yashbhalavat98@gmail.com)
2021-05-13 - 10:08:37 PM GMT- IP address: 66.249.80.199

Document e-signed by Yash.k.Bhalavat (yashbhalavat98@gmail.com)
Signature Date: 2021-05-13 - 10:09:43 PM GMT - Time Source: server- IP address: 72.178.109.5

Agreement completed.
2021-05-13 - 10:09:43 PM GMT