

Senior Design

UConn
SCHOOL OF ENGINEERING



COMPUTER SCIENCE
& ENGINEERING
DEMONSTRATION DAY

GAMPEL PAVILION, STORRS, CT MAY 1, 2015

Senior Design Instructors for 2014-2015

Steven Demurjian, Ph.D.

Swapna Gokhale, Ph.D.

Song Han, Ph.D.

Ion Mandoiu, Ph.D.

Thomas Peters, Ph.D.

Therese Smith, M.S.

Bing Wang, Ph.D.

Contact Information

Reda Ammar, Ph.D., Professor, CSE Industrial Liaison
and Senior Design Coordinator

Reda@engr.uconn.edu

Alexander Shvartsman, Ph.D., Professor and Department Head

aas@cse.uconn.edu

Laurent Michel, Ph.D., Associate Professor and Associate
Department Head

ldm@engr.uconn.edu

University of Connecticut, School of Engineering

Computer Science and Engineering

371 Fairfield Way, Unit 4155

Storrs, CT 06269-4155

Telephone (860) 486-3719

Facsimile: (860) 486-4817 Web: <http://www.cse.uconn.edu/>

CSE Senior Design Course

CSE4939W and CSE4940 are the basis for a year long Computer Science & Engineering Design sequence and capstone project required for all CSE and CS majors that began in Fall 2012. In this sequence, the students embark on a realistic project taking it from requirements analysis to design, implementation, testing and production release. Each project is staffed by a team of 3-6 students working under the guidance of a faculty advisor. Projects will employ either waterfall or agile design methodologies and will make use of modern technologies spanning database, real-time operating systems, simulations, optimization, and computer graphics to build software that runs on mobile devices (e.g., iOS, Android), laptops, desktops or client-server architectures via web-based applications.

Students are expected to learn team management skills, project design skills and demonstrate their skills with software authoring. They experience the realities associated with real-life software design and development. Successful teams design, build and deploy software and documentation that is at least of beta-level quality by the completion of the second semester.

A special thanks to our sponsors:

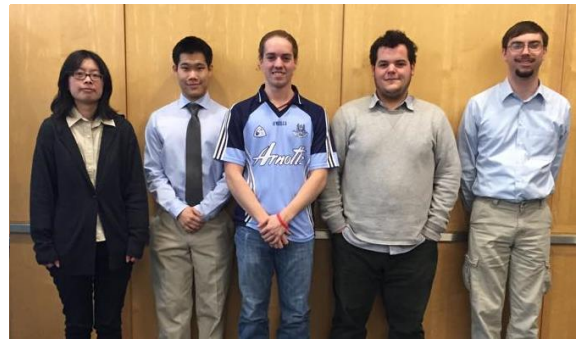
Academic Keys LLC, Belcan Engineering Group, Inc., Connecticut Corsair LLC, Emerson Process Management, IBM, Logicbroker Inc., Pratt & Whitney, Rogo Distributors, Smpl Bio LLC, and the University of Connecticut's Forestry and Plant Science and Landscape Agriculture Departments.

Team #1: Route Bidding Solution

Sponsored by: Rogo Distributors

Sponsor Advisor: David Heller

Faculty Advisor: Bing Wang



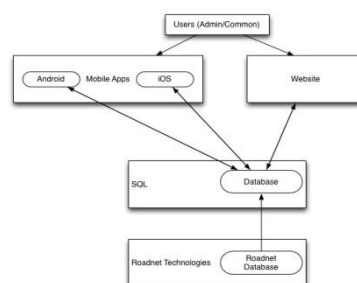
From left to right: Vinkei Wong, Hardy Chen, Matthew Bruzik, Maxwell Madonna, Robert Kowalczyk



The goal of the project is to design and implement Route Bidding Solution (RBS) that streamlines the bidding process for the truck drivers at Rogo Distributors, a beverage delivery company. Currently, truck drivers at Rogo manually pick their routes based on seniority every morning, which is cumbersome and time consuming. RBS allows the drivers to conveniently pick their routes electronically the night before, through a mobile application or the web. It saves time for the truck drivers the next morning. More importantly, it provides the benefits that a driver can be assigned a fixed truck that he drives every day, leading to more responsible behavior and better care of the trucks. In addition, the streamlining provided by RBS allows data to be more conveniently archived and mined to further improve delivery efficiency.

The scope of this project involves pulling data from Roadnet, a commercial routing application, displaying the route data to the user through a user-friendly interface, and sending back to Roadnet the drivers' picks. Another major functionality of RBS is that in the event that a driver cannot or does not want to choose a route on his own, there will be an option to auto-draft. With this, the driver can select some route preferences, and the application will automatically prioritize and choose the best available route based on that criteria.

RBS spans across multiple platforms, including web, iOS, and Android. It is implemented using Objective C, Java, and HTML/PHP for the iOS, Android, and web apps, respectively. A SQL server is used as an intermediate medium for route data from Roadnet. It populates the fields on the website as well as those on the iOS/Android app.

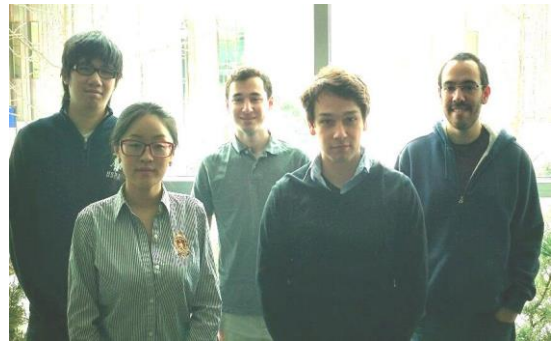


Team # 2: IBM Blue Rideshare

Sponsored by: IBM

*Sponsor Advisors: Konrad Lagarde, David Leip
and Amy Travis*

Faculty Advisor: Thomas Peters

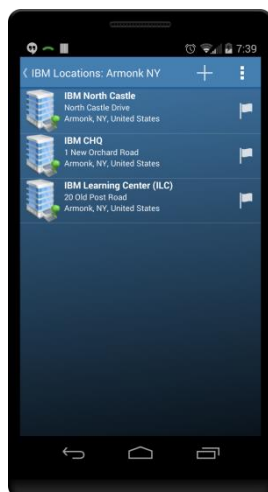


*Front left to right: Da Young Kim, Colby Scotta
Back left to right: Jeff Luong, Conor Finnegan,
Eric Luria*

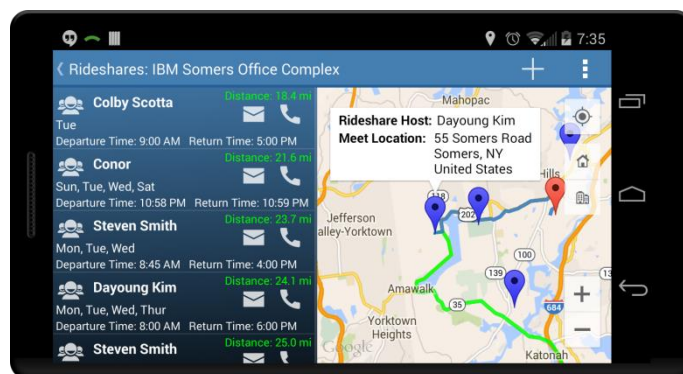
The Blue Rideshare application for Android devices is designed to aid IBM workers in carpooling to work every day. Carpooling to work has many benefits: it saves gasoline, promotes social networking, and provides a convenience for workers who may usually rely on other means of transport. With over 400,000 employees worldwide, it is extremely probable that many of these workers travel similar routes to work each day. The intent of our application is to ease the carpooling process by providing a simple user interface supported by the robust backend services offered by IBM Bluemix.

Our application works by allowing users to create and search for “Rideshares,” or carpool events to their workplace. Any user who wishes to carpool to the same location can view the logistics of the Rideshare, such as the host (person who is driving), meeting times, and distance from their home. Additionally, we provide a mini-map as a visual aid in deciding which Rideshare is most convenient. After choosing an appropriate Rideshare, buttons are available to easily call or send an email to the host expressing a desire to carpool.

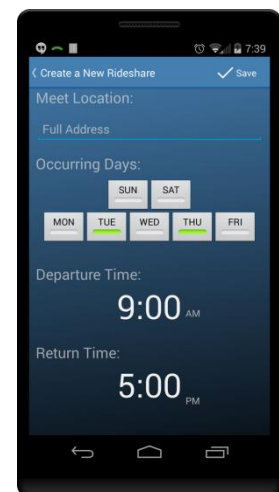
Overall, the goal of our application is to provide a simple medium through which IBM employees can contact one another and make carpooling arrangements. The intuitive user interface makes the process easy and eliminates the stressful overhead of trying to coordinate sharing a ride to work.



Office listings screen



Rideshare listing for an office in Somers, NY



Rideshare creation screen

Team #3: AcroFinder

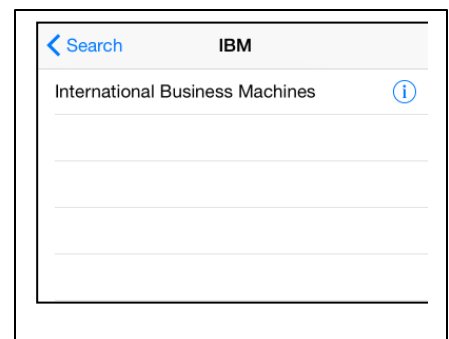
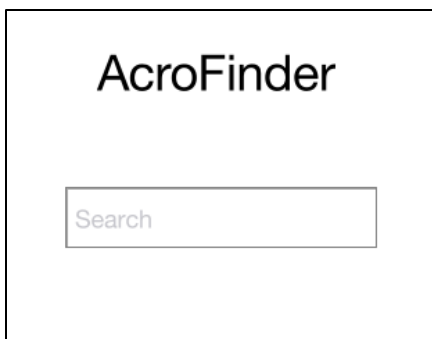
*Sponsored by: International Business Machines
Sponsor Advisors: Konrad Lagarde, David Leip,
and Amy Travis
Faculty Advisor: Thomas Peters*



*From left to right: Krupa Nesadia, Sherwin Yu,
Michael Byon, and Axcel Duarte Espinoza*



AcroFinder is an iOS application developed through the new Apple programming language, Swift. It is a simple way for users to search and look up the meanings to specific acronyms related to International Business Machines (IBM). Easily allowing the user to search the meanings of different acronyms, AcroFinder also provides various in-depth descriptions of the searched item(s). Two different versions of this application were implemented, one that allows only IBM employees to make additions to the acronym word bank and the other that offers only general public usage. This application also comes with user-friendly options that include the ability to “favorite” certain acronyms for an easy-to-navigate future reference, a quick look of recently searched terms, and the ability to view the Most-Viewed Acronyms of All Time, This Month, and This Week. By using this easy, at hand resource, developers can work more efficiently and productively with AcroFinder.



Team #4: Logicbroker Mobile Packing Application

Sponsored by: Logicbroker

Sponsor Advisor: Jordan Robidas

Faculty Advisor: Swapna S. Gokhale



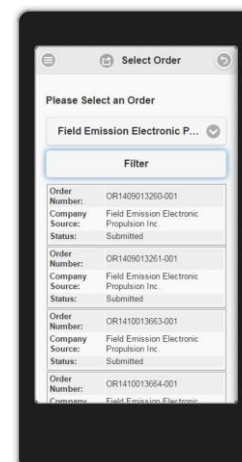
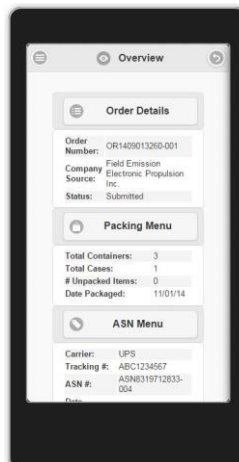
From left to right: Nico Pizzo, Kenneth Philcox, Christian Pinho, Jeff Daniewicz

Logicbroker is a cloud-based software system developed, and owned, by the Logicbroker Company. This “software as a service” automates the many tasks involved in successfully running a company’s eCommerce system. Whereas in the past performing these tasks manually may have sufficed, the pace of today’s market requires a reliable solution that can perform tirelessly and without human error. Logicbroker provides its clients with solutions necessary to keep up with the modern pace of their customers’ demand.

Logicbroker software enables a trading partner to use EDI (Electronic Data Interchange)- a communication standard that allows the transfer of electronic data between multiple companies without ambiguity. This translation of a company’s data standard to EDI and back occurs under the hood. On the front end, retailers and suppliers interact with Logicbroker’s online portal- a visual interface for order management. Retailers are able to receive and fulfill orders, while trading partners can view, acknowledge orders, and prepare for shipping. The functions implemented by trading partners have been the focus of our work.

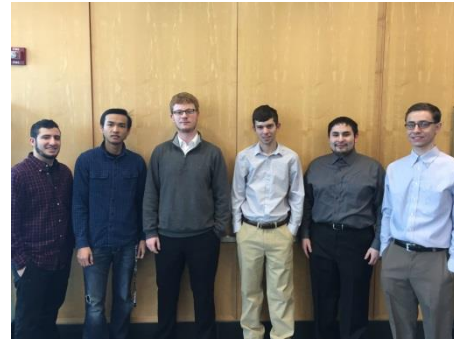
The goal of our project is to improve the user experience of trading partners. To achieve this, we developed a mobile application that synchronizes practical portability with the functionality of Logicbroker’s portal. Ideally, trading partners are able to ship products as quickly as they receive orders from their retailers. As it currently stands, someone in the warehouse must package the order and then use a desktop computer to enter exactly how the order was packaged into the Logicbroker portal before shipping. The mobile packing application assists in a natural, more practical, process by eliminating a constant warehouse to office trip.

Our application is built from two main components, the Logicbroker API and PhoneGap software. Logicbroker’s API is a recent development that allows our application to send, or retrieve, data from the existing portal. This means, in addition to developing this application, we have also been working with them to test, and improve, their technology. PhoneGap is a mobile development framework that allows developers to apply existing web development skills to create and deploy mobile applications- for Android, iOS, Windows Phone, and more- using a single, universal, source code package.



Team #5: Belcan Relational Database: Information Management Tool

*Sponsored by: Belcan Corporation
Sponsor Advisor: Shaila Kambli
Faculty Advisor: Swapna S. Gokhale*



From left to right: Michael Milone, Sokhouth Chay, Kyle Reing, Matthew Bruhin, Nicholas Eng, Eric Boivie

As companies have technologically matured, the capability to gather and organize exceedingly large volumes of data has become imperative. Many companies are currently in a transition period where their data storage and analysis techniques hinder their abilities to make the necessary connections between their data. In order to maintain a competitive advantage, it is increasingly the case that companies need to automate the tasks that are performed manually and present the data in a compact manner.

The goal of our senior design project is to provide Belcan Corporation a tool to gain such a competitive advantage through automation of data management, analysis and reporting. The department within Belcan that our team is working with needs to keep track of software test data associated with Pratt & Whitney jet engines. There are three features that are essential to successfully handle the requirements set by this project. These include data consolidation, ease of access, and insight gathering.

To effectively consolidate data, our job is to provide seamless import functionality that takes currently existing spreadsheet documents and inserts them into a universally accessible database. For this requirement, it is crucial for the software to be able to adapt to multiple document formats. In order to preserve the integrity of the data, checks need to be made to ensure the imported documents align to a set of predetermined specifications. Ease of access encapsulates the ability to intuitively interact, view, and edit the data. This is accomplished through a web-based GUI accessible from Belcan's intranet. We gather insights to build relations across data that was previously stored in disparate spreadsheets. These relations may then be used to generate reports from dynamically selected criteria. This is important because it allows Belcan to make informed decisions based on concrete data that was not readily available.

The tools that our team has opted to use include Visual Studio and Microsoft SQL Server. Both Visual Studio and SQL Server were chosen due to their synergy with the ASP.NET framework. Within this framework, we have opted to use the MVC design pattern to structure our code.



Team #6: Cloud-Based Transcriptome Quantification from RNA-Sequencing Reads

Sponsored by: Smpl Bio LLC
Faculty Advisor: Ion Mandoiu



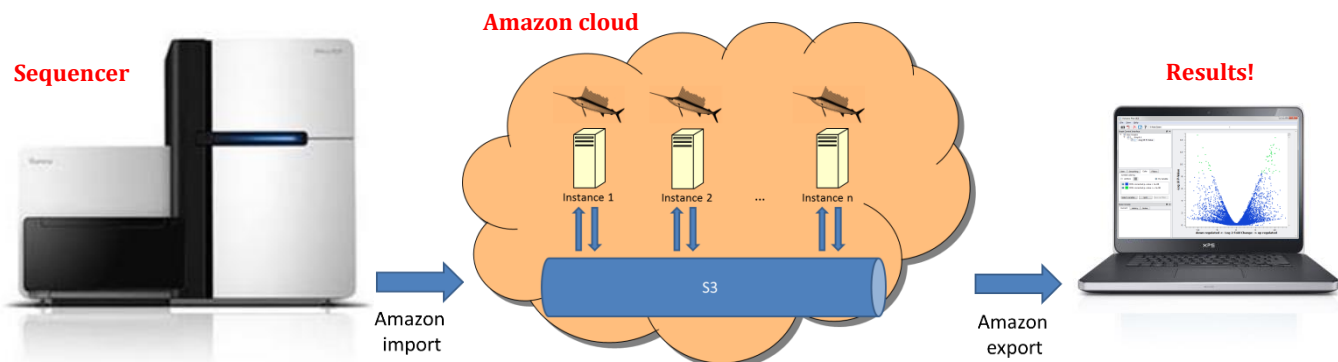
From left to right: David Chen, Nicholas Kruczek, Reynaldo Morillo, Matthew Bemis and Zachary Dicesare.

Smpl Bio LLC is a UConn spin-off based in Tolland, CT. The company is focused on delivering powerful yet easy to use bioinformatics solutions for targeted genomics, including best-in-class tools for biomarker selection, experimental design, and data analysis. These tools are delivered to users within a Software as a Service (SaaS) framework hosted on the Amazon Elastic Compute Cloud (EC2). The system is designed to leverage the scalable nature of cloud-based computing for performing complex analyses quickly and reliably. Access to the system is provided to researchers through a representational state transfer (REST) application program interface (API) server.

The Senior Design team composed of Matthew Bemis, David Chen, Zachary Dicesare, Nicholas Kruczek, and Reynaldo Morillo, five Computer Science/Computer Science and Engineering students, worked under the supervision of faculty advisor Ion Mandoiu with Smpl Bio software engineers James Lindsay and Mike Lydon on implementing a cloud-based transcriptome quantification API that will allow researchers to apply Smpl Bio's algorithms starting directly from RNA sequencing (RNA-Seq) reads.

In the first phase of the project the team conducted interviews with RNA-Seq users across the UConn campus and distilled their feedback into a set of requirements. They also performed a preliminary evaluation of existing RNA-Seq quantification tools, selecting for their implementation the recently published Sailfish algorithm. Sailfish focuses on speeding up the RNA-Seq expression estimation process without sacrificing accuracy. The algorithm operates on a set of reference transcript sequences and RNA-seq reads given in fastq format. Reads are analyzed using a two-step process consisting of read and transcript indexing followed by quantification via Expectation-Maximization. The most salient feature of Sailfish is the replacement of the read mapping step used by other methods with very fast k-mer indexing and counting algorithms based on hashing. This allows Sailfish to quantify transcript abundances over 20 times faster than other existing tools.

In the second phase of the project the RNA-Seq quantification API based on Sailfish was implemented and deployed on the Amazon EC2. The implementation was done in Python and was built using the powerful Celery distributed task queue system for handling asynchronous tasks and scheduling. Algorithms developed during this project will be integrated within Smpl Bio's SaaS framework to ensure high availability and scalability and provide users access without costly investments in high-performance computing infrastructure.



Team #7: F4U Corsair Aircraft Flight Simulator

*Sponsored by: Connecticut Corsair
Sponsor Advisor: Craig McBurney
Faculty Advisor: Bing Wang*



*From left to right: John Foster, Leon Gunzl,
and Julian Schwartz. Not pictured: Alexander
Dow, Sandra Hamilla*

The CSE team's ultimate goal is creating an authentic simulator for F4U Corsair that will respond to user input via the airplane's controls. This is part of a larger multidisciplinary project that aims to renovate and design the simulator for F4U Corsair, an American fighter aircraft that served in World War II and the Korean War.

The CSE design team is in charge of creating the simulated environment that the user will experience inside the cockpit. Prepar3D will be used to simulate the view. Prepar3D software development will be done through SimConnect SDK and add-ons can be developed through the use of various languages such as C, C++, C#.net or VB.net. The Prepar3D software contains preloaded settings that will be simple to adjust to our requirements for development and contains options to add features to the simulation such as custom missions, weather manipulation, control view, and or other artificial intelligence (AI) aircrafts.

The CSE team is currently planning on creating three modules within Prepar3D to target three different audiences. The audiences include children, attracting donations, and appealing to veterans. The module pertaining to amateur flyers, such as children will take the user on a flight programmed to work with limited input to start the plane, and then the program will take control to continue with a predetermined flight pattern of small maneuvers and a landing. For the remaining modules, added user controls will be implemented. Those interested in the project will be able to start the plane and have the capability to control several small maneuvers and complete a landing. This will allow them to experience the input instruments and the reactions of the plane simulation to these inputs. For veterans, the simulator will allow for complete control of the input and the flight patterns. In addition, the veteran's module will have a few combat interfaces and AI to mimic experiences from World War II.

Finally, the team has been given the objective to plan, and if time allows, create a vision system for the simulator. There have been many different forms of displays for simulators predating our own, so research has to be done to choose the display to best fit our needs. The only display option that is being looked into at this moment is immersive display which is a domed shape canvas that allows for the complete "immersion" of the pilot in the simulated environment.

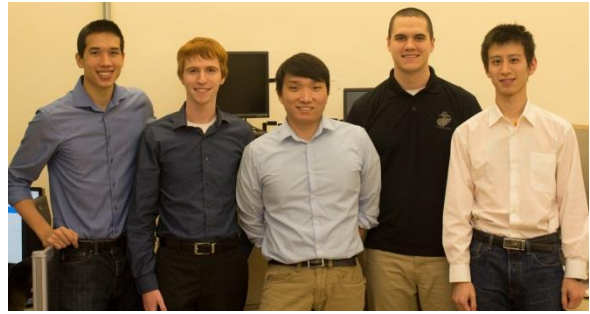


Team #8: WirelessHART on IEEE 802.15.4e: Feasibility Analysis, Implementation & Evaluation

Sponsored by: Emerson Process Management

Sponsor Advisor: Eric Rotvold

Faculty Advisor: Song Han



From left to right: Marcus Li, Timothy Barron, Duy Bui, Tyler Plude, and Tao Gong



WirelessHART, the first international industrial wireless standard (IEC 62591), is built on top of the IEEE 802.15.4 standard, which defines the physical layer and data link layer for low power, low data rate, personal area wireless sensor networks. Both standards have progressed since the WirelessHART incarnation. WirelessHART has gone through a major release, added support for discrete devices, and lately turned attention to real-time wireless control. While WirelessHART is still based on IEEE 802.15.4-2003, IEEE 802.15.4 has progressed to the latest version of IEEE 802.15.4e in 2012, with lots of new features that used to be exclusively in WirelessHART.

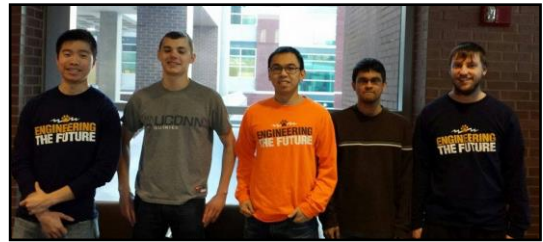
In this senior design project, we are going to study the relationship between these two international standards, and give a feasibility analysis on building the latest WirelessHART standard on 802.15.4e physical and data link layers. We shall develop the WirelessHART communication stack on top of an open-source 802.15.4e implementation, and on energy-efficient embedded platforms provided by two chip vendors. The WirelessHART communication stack will be enhanced with a 6LoWPAN layer to support both WirelessHART and Internet applications. We shall also design and develop a software-based Gateway and Network Manager to manage network resource and regulate real-time traffics in WirelessHART networks. Representative reliable routing and real-time scheduling algorithms will be implemented to satisfy stringent end-to-end timing constraints for safety- and mission-critical applications. A visualization tool will also be developed to demonstrate the real-time network traffic and device health information.

Our performance evaluation will focus on 1) the timing accuracy and synchronization performance of the 10 millisecond time slot on three different hardware platforms, 2) energy efficiency of the communication stack and hardware platforms, 3) the co-existence between WirelessHART networks based on 802.15.4-2003 and 802.15.4e standards respectively, and 4) the efficiency of the resource management protocols proposed in the developed Gateway and Network Manager.



Team #9: Mass Mailer Project

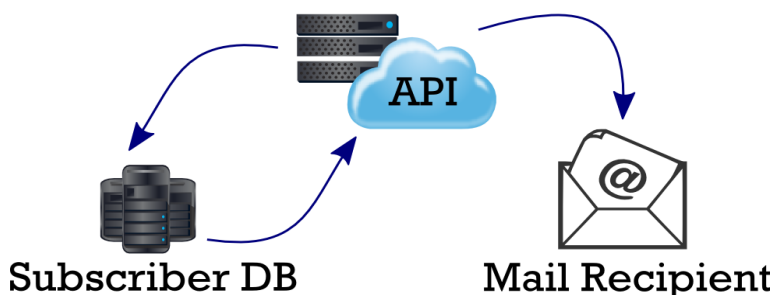
Sponsored by: Academic Keys
Sponsor Advisor: Keith Palmer
Faculty Advisor: Steven A. Demurjian



From left to right: Chao Lai, Christopher Mechler, Jevic Jardeleza, Thanoj Singh, Michael Miller

AcademicKeys.com is the premier source for academic employment and was founded by a University of Connecticut Civil Engineering professor. **AcademicKeys.com's** 17 discipline-focused sites offer comprehensive information about faculty, educational resources, research interests, and professional activities pertinent to institutions of higher education. More than 89% of the top 120 universities (as ranked by US News and World Report) are posting their available higher education jobs with AcademicKeys.com. One of the major computing requirements that is needed in support of **AcademicKeys.com** is the ability to send tens of thousands of batch emails to subscribers organized on different mailing lists. In support of **AcademicKeys.com**, this project has designed and developed **Mass Mailer**, a PHP-based web application that will manage and handle a large amount of mailing recipients each tied to one or more mailing lists. Each mailing list can be modified to add, delete or update any recipient. Furthermore new mailing lists can be created or deleted. In conjunction, users or other applications will have the ability to generate and modify either plaintext or HTML emails and send them to the various mailing lists. This application must be able to handle the ability to send, at the bare minimum, upwards of 60,000 emails per day and store upwards of 1,000,000 mailing recipients. Manipulation of these interfaces will primarily be done through a series of HTTP calls to a REST compliant back-end API.

The ultimate goal of Mass Mailer was to create a REST API that can operate as an efficient mailer that can handle a large amount of data being stored, retrieved and sent. Furthermore, another goal is for the code base to be neat and efficient allowing future upgrades, features, and preventive, corrective, and adaptive maintenance to be a simple and easy process. A model view controller design pattern in conjunction with a robust existing PHP framework has been utilized to support these goals. The Mass Mailer was coded in PHP5 to set up the page content and employed HTML and CSS to create a simple user interface for the ease of users. A comprehensive REST API was designed and implemented to allow information to be exchanged via [JSON](#). All aspects of the REST API are accessed through a series of HTTP calls depending on the way that the information must be used (i.e. modified, retrieved, etc.). Input for applicable HTTP calls is in a JSON format. Output from each call, when applicable, is returned in JSON format. JSON was chosen since its RESTful API's are fast, reliable, and easy to use. [Laravel](#), a PHP web application framework for the development of the MVC web applications, was selected as the framework for front-end views and back-end API.



Team #10: Scann3D Hub

*Sponsored by: University of Connecticut
Faculty Advisor: Bing Wang*

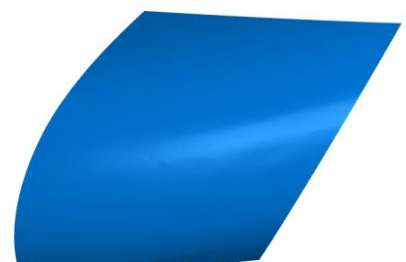
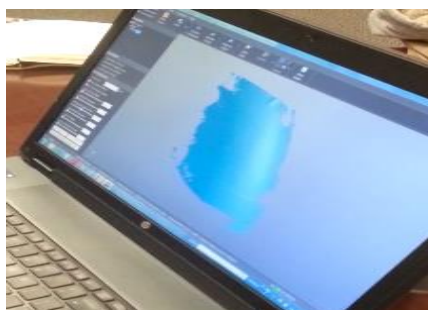


From left to right: Kazi Shahjahan, Dave Ruhlemann, Mitch Thornton, John Guttermuth, Lachezar Shumkov

3D printing is being touted by many as the next industrial revolution. It has already had an immense impact in the manufacturing industry and has grown exponentially in the commercial setting. The goal of this extraordinary research opportunity is to create a flexible database to house 3D “blueprints” for use in both the private and public sector. Users of our program can upload scan files to share with others, creating a growing 3D printing community. No longer will people have to scour the web looking for a specific design when our database acts as a warehouse for users to browse for scan files in order to get the exact print that they desire. The user also has the option to download the scan file locally, allowing for the user to tweak and edit the file to their liking. Another critical element of our product is collaboration. Users possess the ability to edit and share files together easing the design process while stimulating creativity. Never before in such a simple bundle has 3D manufacturing been so easy.

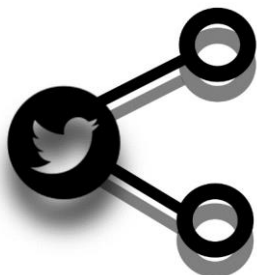
From step one, the primary concern was maintaining simplicity without sacrificing performance or functionality. Using Java3D coupled with creative programming has lead us to develop a natural feeling GUI (Graphical User Interface) that encourages users to act on intuition rather than waste time struggling to find where certain features are located. Through development, we found that Java3D provided a much easier and cleaner way to build the GUI as opposed to using Java Swing. The GUI acts as the bridge between the user and the database, meaning the two need to pair together well. Basing the GUI design off of the key elements in the database, as well as incorporating a Model View Controller and Proxy design style has allowed us to create certain features such as an extremely responsive search menu that updates in real time through Get/Pull requests.

Though there are other competing 3D printing databases, we believe that we have created a product that users and businesses will trust. Certain desirable features such as the ability to share prints privately, collaborate on projects, or even include critical variables such as the temperature or atmospheric conditions at the time of the print are currently lacking in the industry. Our product aims to solve these problems and help grow the 3D community.



Team #11: Twitter Data Acquisition and Analysis

*Sponsored by: University of Connecticut
Faculty Advisor: Swapna S. Gokhale*



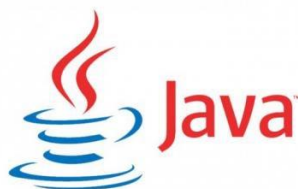
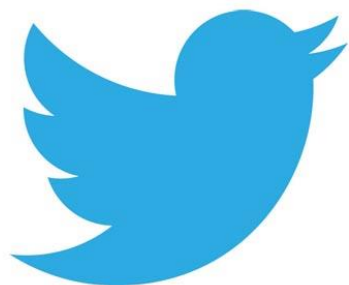
From left to right: Moolchand Ojha, Kimberly Sayre, Russell Bentley. Not pictured: Nicholas Rose

Social networks contain a myriad of information which people may not know that they can utilize. Millions of people sharing their thoughts and perceptions, combined with relevant geographic and network data can give academics and businesses alike a grand new perspective. Data from social networks like Twitter, in conjunction with various machine learning algorithms can be used to see how people are connected, what their perceptions are and even what actions they might take.

The problem is that much of the research is done with quick scripts and other batch style code. This process is neither efficient nor organized. In fact, there is a lot of code that gets rewritten for each project. Tasks like gathering data from Twitter can be time consuming. If researchers were relieved from this overhead they would be able to accomplish more in less time.

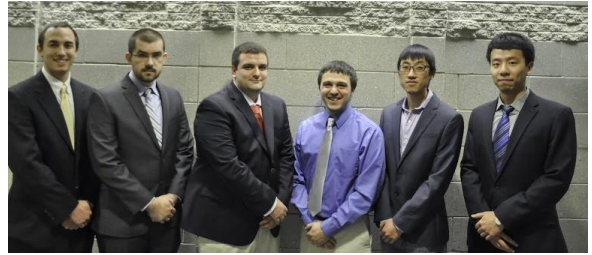
We are building a suite of reusable tools for researchers in order to boost their productivity by reducing their development overhead. These tools include a client for gathering data from twitter, a parser for creating various different datasets for machine learning algorithms, and a network analysis and graphing tool. Throughout the process we are aiming to provide a smooth and intuitive workflow so that researchers can get to their specific interest with minimal effort. However, each tool is also a stand-alone component that can be used however the user sees fit.

The data sets from Twitter are perfect for learning about machine learning as well as performing leading edge analysis. Our hope is that by building and releasing these tools we will be help to further the use of Twitter for academic purposes.



Team #12: Drone Wars - Augmented Reality Drone Dogfight Application

Sponsored by: Team Members
Faculty Advisor: Steven A. Demurjian



*From left to right: Peter Zaffetti, Krystian Charubin,
Michael Grillo, Dustin Peirollo, Chunze Chen, Dake Lu*



As the era of manned warfare draws to a close and unmanned aerial vehicles take to the skies, prepare to take a seat in the command center of a drone. Battle with friends to become an ace pilot by taking out enemy drones in the epic new game of Drone Wars, the augmented reality drone dogfighting app for Android. Drone Wars utilizes unmodified Parrot AR.Drone 2.0 and their on board front-facing 720p HD video camera in combination with an Android smartphone to simulate aerial warfare. With Drone Wars' integrated servers, playing and competing with friends can be done anywhere. Earn the rank of Top Gun by becoming the top ranked player. Players can track their statistics like wins, losses and shot accuracy to compete for total dominance of the skies in Drone Wars' global leaderboard. The setup is as simple as turning your drone on, launching the Drone Wars application and selecting your drone. The server utilizes two wireless interfaces to force drones to connect as clients and then pair them each players corresponding smartphone. Thus, all the drones share a local wireless network to route all flight and video data between paired devices.

The game consists of having the players use their Android devices to control their drones. While controlling their drones, players need to line up the aiming reticle with an enemy drone. The aiming reticle and other game dynamics are overlaid on top of the video input from the camera. Tapping the fire button when the opponent's drone is correctly targeted will result in the targeting player's virtual ammunition count decrementing and the targeted player's life count decreasing by one and allowing them a small window of time where they are invulnerable. The game ends when either the game time runs out or there is only one player with at least 1 life left, who will be the winner. The amount of game time and the number of lives per player is determined at the beginning of the game by the administrative player.



Team #13: Integrated Pest Management App

Sponsored By: UCONN Plant Science and Landscape Agriculture

Sponsor Advisor: Donna Ellis

Faculty Advisor: Steven A. Demurjian



From left to right: Gregory Rendeiro, Ricky Jia, Milod Kazerounian, John Chow, Dillon Shea



The Integrated Pest Management (IPM) App for plant pests provides information that agricultural producers need to address pest problems and assist with pest-management decision making to reduce pesticide applications that may impact flora, fauna, groundwater, and human populations. This has been accomplished by allowing for identification of key plant pests that may cause crop losses or reductions in aesthetic quality to ensure effective directed action against them and providing preemptive knowledge of possible outbreaks by charting the movement of pests, blights and disease across the state, region and nation.

The IPM App for Plant Pests is an application for Android powered devices that allows direct access to a database of information on plant pests in the field. The application helps the user identify plant pests by using photo comparison between damage found on crops and pictures of known infestations, while simultaneously providing resources about the pest and how to manage it using photos, videos, text, and links to web resources. IPM provides information on pests found in Connecticut, with the plan to eventually expand to other states; the application could grow to include the entire United States with enough time and effort once the app is successfully implemented in Connecticut. In order to accomplish this, the IPM App has been designed and executed with state differentiation in mind to easily include additional states and their specific crops and pests. IPM also uses a searchable database of most pests based on category selection such as crop types (vegetables, fruits, ornamentals, herbs, turf, etc.), geographic location and area of plant affected (leaves, stem, roots, fruit). The database also provides photos of the various types of damage for comparison purposes; a user can simply search for pest information by comparing crop damage symptoms and photos of possible pests to explore.

Categories

Vegetables
Plants in the vegetable category.

Fruits
Seeded fruits.

Greenhouse
Greenhouse-grown plants.

Nursery
Nursery-raised plants.

Blueberry

Herbs

Holiday

Perennials

◀ Back

Home Personal Survey Feedback About

Bugs

☐ Beetle

☒ Maggot

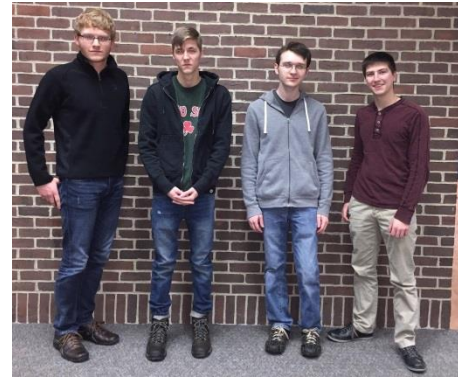
☐ Caterpillar

☐ Moths

☒ Aphids/Mites

Team #14: Parkshark Mobile App

*Sponsored by: Team Members
Faculty Advisor: Steven A. Demurjian*

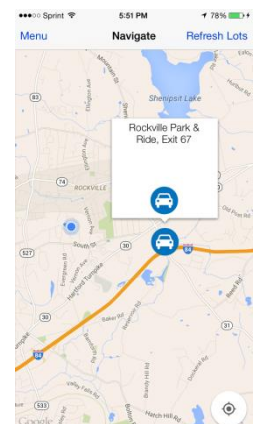


*From left to right: Cameron Panagrosso,
Steven Gerhard, Steven Grasso, Justin
Timmons*

The ParkShark system was developed in order to address the issue of commuter parking on the University of Connecticut's Storrs Campus. Commuters travel between 1-3 parking lots per day, sometimes several times per day; during peak hours this creates unnecessary extra traffic on the already crowded roads, and can cause the commuter to be late to class if there are few remaining spaces. ParkShark is an application system which is intended to shorten the time that users spend trying to find a parking space. The main purpose of the system is to provide a convenient service to UConn's Commuters - reducing the time it takes to get to where the user needs to go and reducing traffic generated by vehicles searching for parking spaces.

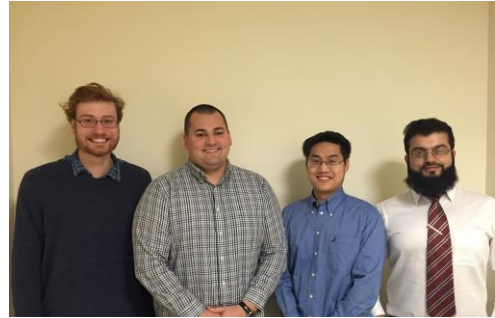
The ParkShark system is comprised of several main components which allow it to operate: a network of small sensors, a database to store pertinent information, and user interfaces on multiple platforms. Each sensor is contained within a weather-proof enclosure in order to survive tough winter conditions and abuse. Additionally, the sensor is built to be low-powered and to recharge itself via solar power - thus minimizing the replacement frequency and maintenance cost. The sensor registers vehicles entering or exiting the lot through the use of a magnetometer; once a vehicle is registered, a message containing lot data and the number of cars involved is sent to the server for processing. The server will update the database accordingly, which is then used to convey information to the user appropriately.

The user interfaces are available on the Android and iOS mobile platforms, as well as a web application for administrative use. Through the use of the mobile apps, a user may visualize the current status of each parking lot, and be routed to the nearest convenient parking lot to the location that they are attempting to reach. The administrative application gives the user the ability to manage parking lot data and user data, including but not limited to: parking lots available on campus, parking lot capacities, permitted parking permits for each parking lot, lot closures, and permitted administrative users.



Team #15: Stormwise Mobile App

*Sponsored by: UConn Forestry
Sponsor Advisor: Thomas Worthley
Faculty Advisor: Steven A. Demurjian*



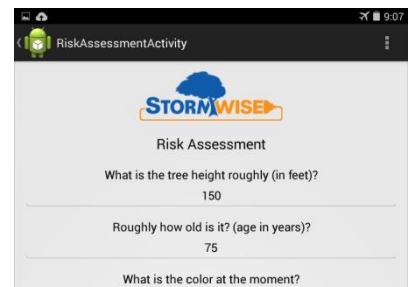
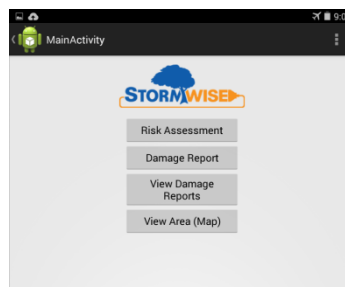
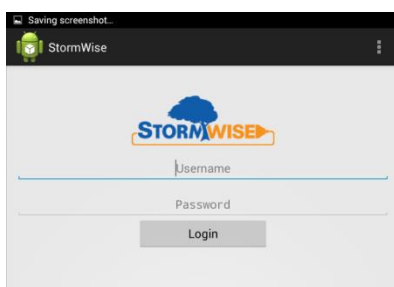
From left to right: Kirk Gardner, Andrew Debarge, Jonathan Chi, Ammad Shakih



The STORMWISE program at UConn (<http://www.stormwise.org>) is intended to mitigate damage associated with power outages caused by large storms (hurricanes, snow storms, thunderstorms, etc.). A successful STORMWISE initiative needs an improved understanding of patterns and modes of tree failure. Since tree structure is complex and highly variable, the establishment of a competent tree failure model has proved elusive. Currently, since we have a poor ability to overcome such complexity, it is vital to build a data collection tool to advance our understanding and knowledge of tree structure, stability, modes of failure, and biomechanics. Data collection fields have been developed in collaboration with Connecticut Light and Power to optimize the speed of data collection. The Stormwise app makes it easy to input data about the fallen tree including manual measurements, images taken on the device's camera, and location based on GPS. The user interface (UI) of the Stormwise app for tree assessment has been designed in order to collect the pieces of information about the tree that are most important to report and which are helpful, but not necessary.

The tree risk assessment component of the mobile app is focused on the user who would like to do a self-assessment of a tree. This component of the Stormwise app adheres to the International Society of Arboriculture's Best Management Practices for Tree Risk Assessment and provides the questions and guidance (including knowledge briefs) needed to perform assessment of risk that allows users to reach a much greater level of awareness of the risk of a tree. Users have the ability to report their knowledge level regarding trees and tree structure. Subsequent questions or data fields are dynamic and flexible by asking more detailed information from experts. Data is marked with user knowledge level when analysis of the data is conducted. The assessment of tree risk has become more precise, but it is largely unknown how accurate the assessments are. Tracking the fate of assessed trees can reveal accuracy of assessments so our Stormwise app will take notice of trees users have assessed and recognize if that tree is reported broken in the database.

The Stormwise app is utilized in two ways: to gather information for reporting, and to feed a website that collects this data and pinpoints where these incidents occur with a ticketing system that will give status changes as the situation changes. The interfaces for both the Stormwise app and website will be dependent on which end the current user is on, the app or the website. There will be a log on for town officials for ticket handling, and users will be able to track their tickets. Note, however, the website app is focused towards the town officials and possibly some utility workers as far as needing to manage and attend to issues/tickets.



Team #16: The J1 – A Modular CNC Machine with an Open-Source Web Interface

*Sponsored by: University of Connecticut
Faculty Advisor: Thérèse Smith*



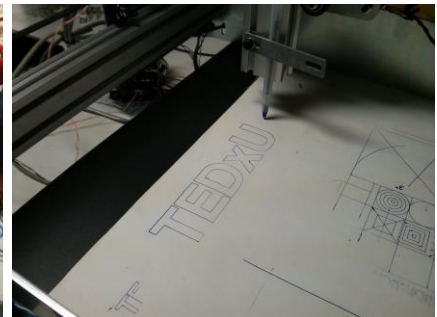
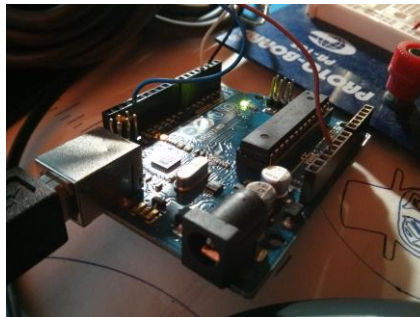
From left to right: Will Percival, Dillon Jones, Nolan Grant, Thérèse Smith, Ryan Powers, Parth Dalsania



Our project is to create a small, modular Computer Numeric Control (CNC) machine with a flexible web-based user interface. Our goal is to enable people with no CNC experience to take advantage of the repeatability and precision of the platform. We have chosen to extend the open-source “Chilipeppr” interface with the ability to draw simple shapes and convert those shapes into instructions for movement, which Chilipeppr can then send to the machine for execution. Our CNC machine is called the J1, and was developed by Dillon Jones with funding from the UConn IDEA Grant. The J1 is able to use the movement instructions (known as GCode) and create sketches using a ballpoint pen, or laser etchings using our own laser diode and optics configuration.

The fall of 2014 brought with it a lot of work researching possible solutions to the problem we want to solve and defining the scope of the project. The specification has set the scope of our project. In it we described the web technologies and the hardware that we plan on using to make your dreams a reality. On the web side of things we have been using a mix of Bootstrap, JSON, and the open source project Chilipeppr to web enable the J1 through the chrome browser. The J1 makes use of a TI Beagle Bone Black (BBB) as a web server, a Sythetos TinyG to handle coordinate interpretation, four stepper motors to drive the gantry, and peripherals.

With all the i's dotted and t's crossed, it's time to use the sum of our engineering educations and create. This semester we plan to finalize designs for the laser etcher and extend Chilipeppr's web interface, which will include designing circuits and PCB's, web development, and use of web-based 3D graphics. Our modifications will enable Chilipeppr to support the creation of simple, machineable shapes directly on the web page. Finally, we will design and implement a testing strategy to objectively qualify the J1 and web interface for use outside our team.



Team #17: Electronic Engine Control Fault Data Analysis

*Sponsored by: Pratt & Whitney
Sponsor Advisor: Jeffrey Grout
Faculty Advisor: Thomas Peters*

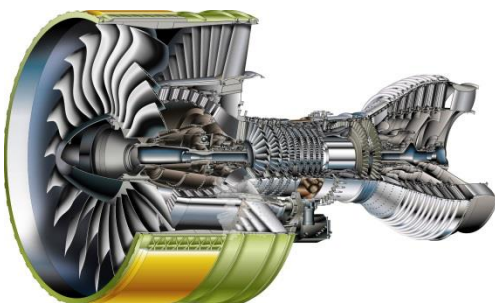


From left to right: Simi Hartstein, Regi Zhang, Kyle Bussmann, Matt Verderame, Jeff White, and Ethan Sperry



Our group was tasked with the creation of an application to analyze Electronic Engine Control (EEC) fault data for Pratt & Whitney's commercial airline engines. This data is recorded by the EEC whenever the plane's sensors detect something out of the ordinary. Analysis of this data is critical to discovering the source of engine defects, but the raw data is completely unintelligible on its own. Our application will allow the user to leverage powerful tools to view and analyze this engine fault data.

Due to the sensitive nature of the data being handled our application will not be displayed at Gampel Pavilion.



©2014 ENGINE ALLIANCE

Fault ID	Timestamp
1	11/10/14 12:00 AM
2	11/10/14 12:00 AM
3	11/10/14 12:00 AM
4	11/10/14 12:00 AM
5	11/10/14 12:00 AM
6	11/10/14 12:00 AM
7	11/10/14 12:00 AM

Name	Value	Value2
Engine Run Time	0	1
Time Since EEC Reset	0	1
Total Operating Time	0	1
Date	0	1
Time	0	1
Aircraft Type	0	1
Aircraft Serial Number	0	1
Flight Leg	0	1
Phase	0	1