STUDENT TECH LAB: LEADERSHIP AND DESIGN (CIST 2010)

The sample projects below were generated using a large language model by analyzing the syllabi from courses students completed in their first semester. Based on past experiences, we know that it can be challenging for partner companies to understand the skills students possess early in their academic journey. To support idea generation, we created this menu of examples. You are welcome to adapt these samples to fit your organization's needs and context.

The projects are organized in three topic areas: Computer Science, Cybersecurity, and Statistics.

COMPUTER SCIENCE PROJECTS

CS.1: Project: Data Cleaning and Analysis for Small Business Sales

Project Description

Students will assist a local business by cleaning, analyzing, and generating insights from their sales data (e.g., CSV files from a point-of-sale system). This helps the business identify trends and customer behavior.

Skills Used

- File handling
- Lists and dictionaries
- Loops and control structures
- String parsing and manipulation
- Functions and modularity

Deliverables

- Python script to read and process raw CSV data
- Cleaned and formatted output file
- Summary statistics report (e.g., top-selling products, total sales per month)
- README file explaining the code and how to run it

CS.2: Project: Employee Directory and Search Tool

Project Description

Students will create an interactive tool for a company's HR team to manage and search an employee directory stored in a text or CSV file.

Skills Used

- Dictionaries for storing employee data
- Functions for modular development
- File I/O
- Conditional logic and loops

Deliverables

- Script for adding, updating, deleting, and searching employee records
- Log file of all changes made
- Documentation for HR usage

CS.3: Project: Customer Feedback Analyzer

Project Description

Analyze customer feedback text to identify frequently mentioned words or phrases and sentiment (basic level). This helps companies understand customer satisfaction and areas for improvement.

Skills Used

- String manipulation
- Lists and dictionaries
- File reading and writing
- Looping structures
- Optional: set usage for uniqueness

Deliverables

- Python script for text parsing and analysis
- Output file with word frequencies
- Visual (text-based or word cloud)
- Summary interpretation

CS.4: Project: Automated Daily Report Generator

Project Description

Design a script for a company to automate the generation of daily operational reports (e.g., compiling log files into structured summaries).

Skills Used

- File I/O (read multiple files)
- String and list operations
- Functions and control flow

Deliverables

- Automated script to extract and compile data
- Sample input and output files
- User guide or runbook

CS.5: Project: Mini Inventory Management System

Project Description

Build a simplified inventory tracking tool where staff can view, add, and modify stock items using Python. Suitable for small retail or storage operations.

Skills Used

- Data structures: lists, dictionaries
- Loops, conditionals, and functions
- File saving/loading

Deliverables

- Interactive command-line script
- Persistent inventory storage file
- User documentation

CS.6: Project: Appointment Scheduler for Local Clinics

Project Description

Develop a simple scheduling system that allows clinic staff to add, view, and cancel appointments, ensuring no double-bookings.

Skills Used

- Lists and dictionaries for data storage
- Conditional logic and control flow
- Functions for modular design
- File I/O for persistent data

Deliverables

- Command-line scheduling application
- Sample appointment database file
- Documentation for setup and use
- Test cases to validate functionality

CS.7: Project: Student Grade Tracker for Tutors or Educators

Project Description

Create a tool to help individual tutors or small educational centers track student grades and calculate averages, attendance, and progress over time.

Skills Used

- Lists, dictionaries, and file processing
- Control structures and functions
- String formatting and manipulation

Deliverables

- Interactive script for entering and retrieving grades
- Summary report with performance insights
- CSV files with student records
- User instructions and interface guide

CS.8: Project: Simple Expense Tracker for Freelancers

Project Description

Build an expense-tracking program to help freelancers monitor their spending across different categories and time periods.

Skills Used

- Data structures: dictionaries for category tracking
- File I/O
- Functions and input validation
- Arithmetic operations and formatting

Deliverables

- Command-line program to add/view expenses
- Monthly or category-based summary report
- Data file with expense logs
- Help guide for non-technical users

CS.9: Project: Basic Chatbot for Common Customer Questions

Project Description

Design a rule-based chatbot that can respond to frequently asked questions for a small business, such as store hours, product availability, or contact information.

Skills Used

- String manipulation and comparison
- Conditional statements and loops
- Dictionary-based response matching
- Functions for modularity

Deliverables

- Scripted chatbot program
- Sample conversation logs
- Response guide for supported queries
- Simple extension file to add/update FAQs

CS.10: Project: Donation Tracker for Non-Profits

Project Description

Create a system for a local nonprofit to track donors, amounts donated, and generate thank-you notes or summary reports.

Skills Used

- File I/O
- Lists and dictionaries
- Functions and string manipulation
- Data formatting and reporting

Deliverables

- Donor management script
- Donation summary report (totals, dates, frequency)
- Template-based thank-you note generator
- Readme and sample data

COMPUTER SCIENCE II PROJECTS

CSCI.1: Project: Local Weather & Alert Dashboard

Project Description

Develop a desktop dashboard that retrieves real-time weather data and displays custom alerts for small businesses or campus facilities to plan activities and communicate with staff.

Skills Used

- File-based alert logging and summaries
- GUI for data display
- Error handling (e.g., internet connectivity)
- Modular code structure for weather API calls and display logic
- Optional recursion for data polling

Deliverables

- Weather dashboard with alert customization
- Live data feed integration
- Source code with usage instructions
- Unit tests and documentation
- Alert simulation scenario pack

CSCI.2: Project: Digital Cookbook with Meal Planning

Project Description

Partner with a wellness center or food pantry to develop a meal planning tool that enables users to create a personalized weekly meal plan based on their dietary needs, budget, and preferred preparation time.

Skills Used

- Classes for Recipes, Ingredients, and Meals
- File I/O for recipe storage and plan exports
- Search/sort/filter logic
- GUI for selecting and previewing meals
- Exception handling for duplicate meals or invalid entries

Deliverables

- GUI-based digital cookbook
- Weekly meal planner with export feature
- Recipe input form and viewer
- Project documentation and annotated code
- Sample recipe database

CSCI.3: Project: Check-in System

Project Description

Collaborate with an organization to develop a system that logs people's check-ins, session durations, and task assignments.

Skills Used

- OOP for student/tutor/session classes
- Time/date handling and sorting algorithms
- File I/O and data backups
- Exception handling and validation
- GUI and version control

Deliverables

- Functional student check-in GUI
- Report generation feature (per day, per tutor)
- Sample usage scenarios with mock data
- Unit testing suite
- Final report or presentation

CSCI.4: Project: Library Self-Checkout Kiosk Simulation

Project Description

Develop a GUI-based application to simulate a self-checkout system for a campus or small community library. Users can check out books, view due dates, and search for titles.

Skills Used

- Object-Oriented Programming (Book, Patron, Transaction classes)
- File I/O for transaction logs and book inventories
- Searching/sorting algorithms
- GUI development for interactive use
- Exception handling (e.g., overdue items, nonexistent books)

Deliverables

- Self-checkout simulation application
- Searchable book database
- Borrowing and returning functionality
- User manual and developer notes
- Version-controlled project repository

CSCI.5: Project: Customer Order Management System for Small Retailers

Project Description

Design a desktop application to help small retail stores manage customer orders, track inventory, and generate reports.

Skills Used

- Object-Oriented Programming (OOP)
- File I/O for order logging
- GUI design (Tkinter or PyQt)
- Searching and sorting algorithms
- Exception handling and input validation
- Version control with Git

Deliverables

- Functional order management desktop app
- User guide and interface documentation
- GitHub repository with version history
- Unit tests for major features
- Sample inventory and orders data file

CSCI.6: Project: Appointment Booking System for Health Clinics

Project Description

Build a system that allows staff to create, update, and delete patient appointments. The system should also generate weekly appointment schedules.

Skills Used

- Custom classes for patients and appointments
- Data structures like lists, dictionaries
- Searching/sorting by date or patient name
- File persistence and backups
- Recursion for appointment conflict resolution
- GUI and basic analytics/reporting

Deliverables

- Application executable or source
- README with installation and use instructions
- Sample test data
- Output files (e.g., schedules, reports)
- Unit tests and change log via version control

CSCI.7: Project: Volunteer Event Coordination Tool for Nonprofits

Project Description

Create a desktop application to manage volunteer signups, event assignments, and availability. The tool should prevent double-booking and include exportable rosters.

Skills Used

- OOP for event and volunteer modeling
- Recursive algorithms for scheduling
- Lists, sets, and dictionaries for data handling
- GUI to input and display data
- Exception handling
- Version control and testing

Deliverables

- Event coordination tool
- GUI screenshots
- Volunteer management documentation
- Exportable volunteer event schedule

- Source code in Git repository

CSCI.8: Project: Expense Tracker with Data Visualization for Freelancers

Project Description

Design a personal finance tracker for freelancers that categorizes expenses, calculates monthly totals, and provides basic visual reports.

Skills Used

- OOP and modular programming
- File handling and CSV processing
- Data structures and algorithm analysis
- Error handling and unit testing

Deliverables

- GUI-based expense tracker app
- Sample data set
- Expense category and trend analysis charts
- Full codebase with inline documentation

CSCI.9: Project: Employee Timesheet and Payroll Estimator

Project Description

Build a tool for small businesses to manage employee clock-ins and calculate estimated payroll based on logged hours, pay rate, and overtime rules.

Skills Used

- File operations for logging work sessions
- Data structures (e.g., dictionaries of employees)
- Algorithm design for payroll and overtime logic
- GUI with entry forms and summary reports
- Exception handling for bad input and overlapping shifts

Deliverables

- Timesheet management GUI
- Payroll calculation report generator
- Sample test files with dummy employees
- Source code and technical documentation
- Version history and issue tracking

CSCI.10: Project: Community Issue Reporting Mobile App

Project Description

Partner with a city department or local neighborhood association to build a mobile app that lets residents report community issues (e.g., potholes, graffiti, damaged signs).

Skills Used

- Computational thinking and algorithm design
- Programming with conditional logic and loops
- Modularization and stepwise refinement
- Mobile application development
- Data storage and abstraction

Deliverables

- Fully functional mobile app (APK or source project)
- Flowcharts or pseudocode for key logic
- Documentation with user guide
- Video demonstration or presentation

CSCI.11: Project: School Supply Inventory Tracker for Nonprofits

Project Description

Develop a lightweight inventory system for a nonprofit that manages and distributes school supplies. It will help staff track donations, inventory levels, and distribution history.

Skills Used

- Data structures (lists, dictionaries)
- Decision-making and logical reasoning
- Programming modular functions
- File/data storage

Deliverables

- Inventory management application
- Database or CSV file for storing supply data
- Admin dashboard or interface
- Project documentation and setup guide

CSCI.12: Project: Interactive Educational Web App for K–12 Students

Project Description

Design a web-based educational tool for a school district to help students practice basic math, reading, or science. The app will present interactive questions and track scores.

Skills Used

- Abstraction and modular design
- Algorithms and control structures
- Web application basics
- Graphics and multimedia integration

Deliverables

- Web app with interactive lessons
- Codebase with modular comments
- Instructor/admin panel (basic)
- Tutorial for classroom use

CSCI.13: Project: Campus Carbon Footprint Calculator

Project Description

Collaborate with UNO's sustainability office to build a mobile/web app that helps students estimate their individual or organizational carbon footprints based on travel, energy use, and consumption.

Skills Used

- Algorithm development
- Logical reasoning and decisions
- Form-based input and calculations
- Application of computing to social good

Deliverables

- Carbon calculator interface (mobile or web)
- Logic documentation and formula breakdown
- Outreach flyer or poster to promote app use
- Final project reflection report

CSCI.14: Project: Healthy Habits Tracker for Wellness Programs

Project Description

Collaborate with a local healthcare provider or wellness initiative to design a mobile or web app that allows users to track daily wellness habits like water intake, sleep, and exercise.

Skills Used

- User input collection and data manipulation
- Control structures (loops, conditionals)
- Simple data structures (lists, counters)
- Mobile app development
- Data visualization (e.g., daily progress bars)

Deliverables

- Habit tracking application
- Data dashboard for daily/weekly trends
- Wireframes or flowcharts of app logic
- User instructions and screenshots

CSCI.15: Project: Digital Literacy Quiz App for Libraries

Project Description

Develop a quiz-style mobile or web app in partnership with local libraries to help patrons assess and improve digital literacy skills (internet safety, email etiquette, etc.).

Skills Used

- Algorithms for quiz logic and scoring
- Loops and decision statements
- Multimedia integration (text, audio, video)
- UI design for clarity and accessibility

Deliverables

- Interactive quiz app with multiple question sets
- Score summary and feedback page
- Administrative panel to add/update questions
- Guide for librarians or facilitators

CSCI.16: Project: Volunteer Hours Logging Tool for Nonprofits

Project Description

Create a tool that allows nonprofit volunteers to log their service hours, view totals, and generate proof-of-hours reports for school or work requirements.

Skills Used

- Structured data input
- Algorithms to track and summarize hours
- Basic authentication logic
- Data abstraction and modular design

Deliverables

- Volunteer portal for logging hours
- Dashboard with running totals and graphs
- Admin panel for review and approval
- Documentation for deployment and use

CSCI.17: Project: Event Planning and RSVP System

Project Description

Help a campus organization or small business manage events by building an RSVP system that collects attendee data and allows event organizers to monitor participation.

Skills Used

- Algorithms for user input validation
- List and dictionary structures for tracking
- File or cloud storage
- Stepwise refinement and modular development

Deliverables

- RSVP form and event dashboard
- Confirmation message and status update system
- User guide and event setup instructions
- Example use case or demo event

CSCI.18: Project: Online Pet Adoption Finder for Animal Shelters

Project Description

Design a web or mobile app for an animal shelter that lists pets available for adoption, with search filters for species, age, and availability.

Skills Used

- Modular application design
- List filtering algorithms
- Conditional logic for search/match
- Graphics and UI considerations

Deliverables

- Pet listing interface with filters
- Backend dataset with pet profiles
- Mock-up site or mobile app
- Usability feedback and summary report

CSCI.19: Project: Local Business Customer Feedback Analyzer

Project Description

Build a web or mobile app for a small business to collect and analyze customer feedback. The tool will use simple algorithms to identify common keywords and ratings trends.

Skills Used

- Data input and processing
- String manipulation and simple data structures
- Algorithms for sorting/categorizing feedback
- Visualization using basic charts

Deliverables

- Feedback collection app
- Text analysis script
- Visual report generator
- Sample reports and user instructions

CYBERSECURITY PROJECTS

CYBR-1: Project: Password Security Analysis

Project Description

Analyze real-world password breaches (from publicly available datasets) to identify trends, weaknesses, and generate recommendations for password policies.

Skills Used:

- Data cleaning and analysis
- Hash cracking basics
- Access control concepts

Deliverables:

- Password trends report
- Proposed password policy document
- Visualization of password strength distributions

CYBR-2: Project: Simulated Phishing Campaign and Analysis

Project Description

Students design a simulated phishing email and deploy it (with permission) within a controlled classroom or club environment to observe user behavior.

Skills Used

- Social engineering
- Risk communication
- Ethical considerations

Deliverables

- Campaign setup documentation
- Behavioral response analysis
- Awareness recommendations

CYBR-3: Security Awareness Campaign for a Local Organization

Project Description

Develop and deliver a campaign (posters, slides, and emails) focused on phishing, password hygiene, and safe browsing for staff or students.

Skills Used

- Communication strategy
- Risk awareness
- Policy framing

Deliverables

- Awareness content pack
- Pre- and post-survey results
- Final impact summary

Some more experienced students will be able to do the project below

CYBR-4: Project: Build a Personal Security Toolkit

Project Description

Each student creates a bootable USB with essential tools for malware detection, file encryption, and network analysis.

Skills Used

- Understanding security tools
- Operating system configuration
- Secure tool deployment

Deliverables

- USB Toolkit
- Tool instruction manual
- Demonstration walkthrough

STATISTICS PROJECTS

STAT-1: Project: Customer Behavior Analysis for a Retailer

Project Description

Students will analyze a retail dataset (such as online purchases or point-of-sale transactions) to identify trends in customer behavior. They will determine the most popular products, seasonal trends, and the impact of discounts or promotions on sales.

Skills Applied

- Descriptive statistics (central tendency, variability)
- Probability distributions
- Hypothesis testing (e.g., comparing pre- and post-promotion sales)
- Regression modeling (e.g., sales vs. discount levels)

Deliverables

- One-page research question and dataset justification
- Two-page progress report with tables and graphs
- Final project report with:
    - Summary statistics
    - Visualizations (histograms, bar charts, box plots)
    - Hypothesis test results
    - Regression analysis and interpretation
- Oral presentation slides

STAT-2: Project: Movie Ratings and Revenue: What Drives Success?

Project Description

Using a dataset from IMDb or Kaggle, students will study how factors like genre, budget, or rating impact a film's revenue or viewer score.

Skills Applied

- Probability distributions (e.g., movie success likelihood)
- Hypothesis testing (e.g., comparing genres or decades)
- Regression modeling (revenue ~ budget + rating + genre)

Technologies Used

- R
- IMDb datasets or Kaggle (Movie Metadata)

Deliverables

- Problem definition and rationale
- Visual analysis dashboard (plots and tables)
- Hypothesis test comparisons
- Regression model output and interpretation
- Report and final class presentation

STAT-3: Project: Social Media Engagement Analysis

Project Description

Students will investigate patterns in engagement (likes, comments, shares) on platforms such as Instagram or Twitter. They can explore how factors like posting time, content type, or hashtags impact user interaction.

Skills Applied

- Descriptive statistics and visualization
- Probability distributions of engagement metrics
- Hypothesis testing (e.g., engagement before vs. after 5 PM)
- Simple regression (e.g., number of hashtags vs. likes)

Technologies Used

- R (rvest for web scraping if needed, ggplot2)
- Publicly available datasets or manually collected samples

Deliverables

- Data collection plan and research question statement
- Summary statistics and visualizations
- Statistical tests and interpretations
- Regression model (if applicable)
- Final written report and class presentation

STAT-4: Project: Ride-Sharing Service Efficiency (e.g., Uber or Lyft)

Project Description

Analyze ride data to understand trends in ride duration, cost, or wait times. Students can assess how these vary by time of day or location and build predictive models.

Skills Applied

- Graphical analysis and time-based trends
- Interval estimation and sampling distributions
- Hypothesis testing (e.g., fare differences across neighborhoods)
- Regression (trip duration as a function of distance, time, etc.)

Technologies Used

- R (lubridate for date-time data)
- Open datasets (NYC TLC trip data, Kaggle ride-sharing datasets)

Deliverables

- Problem formulation and dataset justification
- Time-based graphs and distributions
- Confidence intervals and test results
- Predictive modeling and discussion
- Presentation slides with findings

STAT-5: Project: Analyzing Sports Performance Data

Project Description

Students evaluate athlete or team performance metrics from professional or college sports. For example, they may test whether one team has a statistically significant advantage over another.

Skills Applied

- Descriptive analysis (means, standard deviations)
- Probability distributions of scores or events
- Hypothesis testing (e.g., home vs. away performance)
- Regression (e.g., player performance based on training or time played)

Technologies Used

- R
- Sports datasets from ESPN, FiveThirtyEight, or public APIs

Deliverables

- Statement of performance metrics and research questions
- Visual comparison of athletes or teams
- Hypothesis tests and conclusions
- Regression model with discussion
- Final project report and class presentation