

Suggested Capstone Projects

Semester 1 – 2020

Each group needs to select one of the following projects:

Project 1: Match Making System

You are required to develop a web based matching system, such as matching of jobs required by employers to skills of potential employees. Here are some examples:

1. Potential employers post the details of the kinds of employees they are looking for and job seekers post their skills and experience and the system matches employers to job seekers.
2. Sellers post descriptions of items they have for sale and buyers post descriptions of items they want to buy and the system matches buyers and sellers.
3. Trades people post their expertise and location and householders post details of problems they want fixed or jobs they want done.
4. A matchmaking site.
5. Any other producer-consumer situation of your choice.

The system must:

- Permit users to register
- Provide the necessary transactions for data entry
- Provide appropriate matching algorithms
- Operate from desktops and mobile devices
- Provide appropriate admin functionality

You can add any other enhancements that you think would be useful and appropriate.

Project 2: Smart Plant

Many modern offices keep a number of pot plants to improve the ambience. Unfortunately plants are neglected and die. Acme smart plants is a company that looks after plants in offices. Acme would like to place plants in offices with suitable sensor instrumentation and monitor the health of the plants 24/7. If there seems to be a problem with a plant, Acme will send someone to deal with it.

Your task is to develop a system for Acme. Sensors must be able to check the temperature, humidity and light intensity of the environment and the percentage of moisture in the soil. You need to develop a REST API for Arduino or any other IoT

device that can support the sensors and to develop a web application that can receive the data from sensors and manage plants and customers.

The system must:

- Have a programmed device that is equipped with sensors to collect the data, and send it to the server.
- Save all the data that is coming from sensors for each plant
- Give daily/monthly report/graph to the user about different data of the plant and the state of health
- Provide necessary functions for user, so they can register, login and see their dashboard
- Provide a REST API and a web/mobile friendly app.
- Use Wi-Fi for sending data from the IoT device to web-app
- Provide functions for admin to manage users, plants and any other admin functions

An IoT device must be programmed to collect the data through sensors. For this you can use any hardware, however, we recommend Arduino. Here is a tutorial for using Arduino for this project and some sample code.

<https://www.survivingwithandroid.com/iot-project-tutorial-smart-plant-system/>

Project 3: Car Share Scheme

You are required to develop a web/mobile application for a company running a car share scheme. The company owns a number of cars which can be parked at a number of locations in the city. Users of the scheme will book a car for some period of time, use it and then return it to an empty location (the return location could be different than the book location). The system needs to provide the necessary transactions and data management/storage capabilities to support this kind of business.

The system must:

- Register users
- Keep all car data and rental information
- Provide a way for a user to book (and return) a car for a particular time and location
- Have an option to find the nearest available car based on the user's location
- Provide a report for users to see their past bookings with date, time and location information
- Be platform-independent and work through web, or any mobile platform.
- Have enough dummy data to test user scenarios in different locations and with different cars

Project 4: Learning Analytics Visualisation (LAV)

You are required to conduct an investigation on a publicly available data set of parking events in Melbourne and write a report on your findings.

This folder contains two files:

https://rmiteduau-my.sharepoint.com/:f:/g/personal/vic_ciesielski_rmit_edu_au/Eo-ZMNJzVGBMqW-ZY1H4gcIBXp5lf8hf7gdAZKBNwB2FIA?e=bRIIDG

1. parking-small.csv which contains a small sample of the full data.
2. parking_events.zip, the full data, 250MB compressed, expands to 1.8 GB

You need to think up interesting questions to ask about the data and work out a method for answering them. Think about the kinds of things the owner of the data might want to know from the data. For example you might ask “In which places is it hardest/easiest to get a parking spot?” or “Is it possible to predict whether a particular parking event will result in an infringement”. To answer the questions you will need to extract relevant data from the dataset and prepare it for suitable algorithms.

The report should explain the methodology and software tools that you used and any interesting things that you found. It should include suitable visualisations of the data. Any scripts or programs developed should be included in an appendix.

You can choose other publicly available data to work on. You can find some Melbourne data here: <https://data.melbourne.vic.gov.au/>

Your tutor needs to approve the data files.

Project 5: Live Video Stream and Video Share App

This is a mobile application that lets family and friends share uploaded videos and live streams. Users can upload and share videos of up to 6 minutes and can share live streams through their own channels. You can use any technology for this, however it must be platform independent. You can use any opensource library for sharing the video or livestream, however you need to have your own watermarking algorithm/solution for copyright reasons.

This app must:

- Provide access to public users to only watch the videos
- Let anyone register and be a user
- Let users log in and create their own channels
- Let users upload and tag their videos
- Let users “like” shared videos and add comments
- Let users do live stream
- Enable public comment on live stream videos or send emoji live and at the same time
- Admin functionality for managing users and videos through web access or mobile app

Advanced Feature:

- System must track the videos watched by users and suggest future videos based on their previous topics watched videos.

Project 6: Secure Voice/Text chat server/app

This project is concerned with the development of a secure chat server that can be a platform for project teams. Anyone can join and engage in voice or text chat with others in his/her team through different channels. This is a client/server architecture. The server is the host for all users. The sender uses the client app to send the voice/text to the server, then server redirects the message or voice to the receiver. You may need to choose a communication protocol between your server the client app such as TCP or HTTP Post, based on your software architecture. Users should be able to register and create a voice/text channel for their team so that other teammates can be invited and use their channels in their group area.

The system must:

- Provide a registration function for individuals to join the application.
- Enable users to make groups and invite other users to their group as a group admin task.
- Enable a dashboard, profile and avatar for each user.
- Permit group admin to create different channels for the group and limit the access of group members to each channel.
- Users must be able to chat or voice when they are online and inside the relevant channel.
- The server must be able to handle the traffic of a growing the number of users.
- The text communication must be secure and encrypted.
- Users can add each other as friends and chat directly one on one.

Feature that is nice to have:

- Users can send/upload attachments as documents/pictures/videos to their channels.

Project 7: Online Virtual Wallet App

This application is an online application that helps users to manage their personal finances. The app lets users transfer, pay bills and buy online through their account.

Anyone over the age of 18 and has a credit card can register for this system.

Basic requirement for this system are:

- Secure registration and authentication system
- Users must have a dashboard so they can see their past transactions and their profile, also an avatar
- Users must be able to add money to their online wallet so they can transfer/pay-bill/ online shopping
- Users can send money to each other by using each others user-id
- Users can pay their bills through B-Pay if they can find the company on the platform after entering the company B-Pay reference number.
- The platform must provide an API for online shops so that users can pay their shopping through this facility
- The default currency is Australian Dollar, however any shop in the world can use the API
- System must calculate the currency exchange for other currencies and deduct 1% commission
- Admin functionalities for managing users accounts

Advanced features:

- Users can add friends by their user-id, chat with them and send limited money envelop to their friends through the text chat
- Merchant management functions for online shop owner, so they can view their transactions
- Admin functions for managing merchants.