**A film recommendation system**

**Supervisor**

P. Severi (pgs11)

**MSc Programmes**

Agile Software Engineering Techniques (SET)

Web Applications and Services (WAS)

Advanced Computation Methods (ACM)

Advanced Software Engineering (ASE)

Advanced Computer Science (ACS)

**Prerequisites**

Excellent programming skills

How to develop web based applications

**Aims of project**

 The aim is to develop a system that will generate recommendations of films to watch according to what the users want and  how other users have rated them.

**Challenges presented by the project**

Understand and find out

* different ways to ask users to rate films so that other users can find their own preferences.
* ways to  get users to make pairwise comparisons between films
* rating and matching methods.
* how to present the recommendation information to users.

**Learning outcomes**

Understanding how recommendation systems work.

**Nature of end-product**

A web-based film recommendation system

LightwaveRF API: https://github.com/pauly/lightwaverf Devices: https://lightwaverf.com/ Raspberry Pi: https://www.raspberrypi.org/

Phase 1: Mobile app development.  
Phase 2: User evaluation.  
Phase 3: Dissertation Writing.

### References

[] **Joaquín Torres-Sospedra, Joan Avariento, David Rambla and Raúl Montoliu - etal..**

Enhancing integrated indoor/outdoor mobility in a smart campus, International Journal of Geographical Information Science (2015).

## Web Application for Allocating Groups, Topics and Supervisors for Group Discussions

### Supervisor

P. Severi (pgs11)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Web Applications and Services (WAS)

Advanced Computation Methods (ACM)

Advanced Software Engineering (ASE)

Advanced Computer Science (ACS)

### Prerequisites

Some programming ability. Knowledge about matching and flow algorithms will help but is not absolutely required.

### Aims of project

Create a web application where supervisors can specify the topics that they would be willing to supervise, students can submit their preferences, and an allocation of students to groups, topics and supervisors is produced automatically. This is needed for the Personal and Group Skills module (CO7210). The web application will store the preferences and calculate an allocation of students to groups after the deadline for submissions.  
  
There should be an administrator, who can do things such as the following:  
\* view and edit the preferences of the students (in a table)  
\* run the matching algorithm  
\* view the output of the algorithm (the allocation of the students) and publish or mail it  
  
If the administrator agrees with the allocation, he or she should have the option of making the allocation visible or possibly sending mail to all the students with their allocation. (The students would receive an email with their allocated topic and the names and email addresses of their group members.)  
  
Once the allocation is visible, the students can find it by logging into the system again (or even just going to the website without logging in).

### Challenges presented by the project

Design the website to allow students to select their desired topics from a drop down list. Each student should be made to select (at least) four topics, in order from most to least liked.  
  
Allow the administrator to see preferences, run the matching algorithm, and view (and ideally edit) the output.  
  
Design a simple database which holds the list of projects (names and codes) and preferences for each student.  
  
Implement a maximum weight matching algorithm to match students to groups, under the following conditions:  
\* each group should have 4 or 5 members (as many groups as possible should have 5 members)  
\* each student should preferably get matched to their first topic, then the second, and so on. For instance, the weight of topic 1 of each student could be 4, the next weight 3 etc.  
\* Each student should be assigned exactly one topic, and each topic should be assigned to at most one student (not all topics must be assigned).  
  
The problem could be formulated as a flow problem with weights and capacities on the edges.

### Learning outcomes

Ability to design and implement a web application to address a real-world problem.

Ability to implement matching or flow algorithms and apply them to new problems.

### Nature of end-product

Working web application, written dissertation.

### Project timetable

Month 1: web application and database design  
Month 2: implementation of matching algorithm  
Month 3: debugging, evaluation and dissertation writing.

**Yoga class planner**

**Supervisor**

P. Severi (pgs11)

**MSc Programmes**

Agile Software Engineering Techniques (SET)

Web Applications and Services (WAS)

Advanced Computation Methods (ACM)

Advanced Software Engineering (ASE)

Advanced Computer Science (ACS)

**Prerequisites**

* Programming abilities
* Ability to create a GUI
* web based application

**Aims of project**

An online web application which will allow teachers to plan  
a class knowing the ability and possible physical problems of  
the students beforehand.  
  
The students should be able to book a class,  add their level, abilities  
and fill in a health questionnaire.  
  
  
The teachers should be able to

* add new  asanas (poses), instructions how to do them, teaching points and contraindications
* visualize muscles involved
* classify assanas in certain types (e.g. backbends) and levels (e.g. beginners, modifications and variations)
* search asanas for type, muscles involved, level, etc
* create sequence/flows of asanas

The system should be able to automatically recommend   asanas to add in a sequence  based on the health and preferences of students attending the class.

**Challenges presented by the project**

* Friendly booking of classes using a calendar
* Building sequences/flows of poses
* Automatic recommendation of poses to  add to the sequence

**Learning outcomes**

* Ability to  store  sequences of asanas that could be shared amongst the teachers
* Ability to create a  web application
* Ability to implement an algorithm that recommends poses to teachers when they build sequences/flows for their classes

**Nature of end-product**

A web application to help yoga teachers to build different flows of poses to teach in their classes.

## Takeaway Menu System

### Supervisor

N.Yap (ny50)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Web Applications and Services (WAS)

Advanced Software Engineering (ASE)

Cloud Computing (SCL)

Human Technology Interaction (HTI)

Advanced Computer Science (ACS)

### Prerequisites

Good programming skills, software engineering, database design

### Aims of project

The aim of this project is to develop a web-based system to manage the takeaway service. The system should handle both orders from telephone and website. In addition to that, a rating function could be provided for the services delivered to the customers.

### Challenges presented by the project

Understanding how takeaway works, understanding how to implement an online system to accept both telephone and online orders.

### Nature of end-product

A web-based system for takeaway service.

## Stock Control System

### Supervisor

N.Yap (ny50)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Software Engineering for Financial Services (SFS)

Web Applications and Services (WAS)

Geospational Intelligence (GI)

Advanced Computation Methods (ACM)

Advanced Distributed Systems (ADS)

Advanced Software Engineering (ASE)

Cloud Computing (SCL)

Advanced Computer Science (ACS)

### Prerequisites

Any knowledge in web technologies and database for building a web-based system. A good understanding of stock inventory management system is beneficial.

### Aims of project

This project aims to create a software system to monitor/manage the stock levels of computer components in a computer warehouse.  The system should be able to specify the exact location of each item during the stock assembly process.  The system should also provide some reporting tools and invoice facilities as well.

### Challenges presented by the project

Building a complex software system.

### Nature of end-product

A web-based system for stock control.

### Supervisor

I. Ulidowski (iu3)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Web Applications and Services (WAS)

Geospational Intelligence (GI)

Advanced Computation Methods (ACM)

Advanced Distributed Systems (ADS)

Advanced Software Engineering (ASE)

Cloud Computing (SCL)

Human Technology Interaction (HTI)

Advanced Computer Science (ACS)

### Prerequisites

Web application development skills.

### Aims of project

Create a mobile (or a web) application that allows users to book and pay for car parking spaces. You could include other functionalities inspired by, for example, PayByPhone or other parking apps. The modelling can be done using, for example, the process calculus CCS (taught in CO3007).

### Challenges presented by the project

Understanding modern web application development frameworks.

The purpose of this project is to specify a subset of functionalities of a car parking app, mainly concentrating on exchange of information between the user to the app, and on the flow of interactions involved in using such app.

### Learning outcomes

At the end of this project the student will: be able to translate informal requirements into an concrete design and

implementation; be able to develop moderately complex web applications.

### Nature of end-product

We assume that there is only one parking site. Customers either arrive at the site and try to purchase parking, or they try to book parking prior to driving to the site. In the first case, when they arrive at the site, they enter intended period of parking (start time and duration, all in round numbers of hours) to see if any empty bay is available. Please note that even if a bay has no car parked, it may be booked for another user. In the second case, the user is asked to enter the intended period of parking by providing the date, start time of parking and the duration, all in round number of hours. Again, the system should let the user know which bays (if any) are available, and allow the user to select one.

Users pay for parking or booked parking upfront using credit cards. We assume for simplicity that all credit cards are issued by one bank. Each credit card has a credit limit, which cannot be breached. So, a payment with the app can be declined if a parking charge plus current balance on a card is greater than its credit limit.

**The Drop-Shippers Supplier Database (DSSD)**

**Supervisor**

R. Heckel (rh122)

**MSc Programmes**

Agile Software Engineering Techniques (SET)

Software Engineering for Financial Services (SFS)

Web Applications and Services (WAS)

Advanced Software Engineering (ASE)

Cloud Computing (SCL)

Human Technology Interaction (HTI)

Advanced Computer Science (ACS)

**Prerequisites**

The project requires strong web development skills. A preliminary implementation uses MongoDB, React and NodeJS and Express.

**Aims of project**

Community Enterprise Engine is a social enterprise consultancy helping community venues start trading.  Three directors sell consultancy services including SEO, CRM, websites, training, business, marketing, community development and grant writing services.

We would like to build a platform with an integrated CRM, database, mailer that provides rated dropshippers resellers.  We would also like a ‘nearest supplier’ factor which reduces carbon footprint from transport.  Resellers can be thought leaders, influencers, ecommerce website owners, charities and NFPs or marketeers.

A dropshipper is a supplier that stocks and fulfils your order, so that you as a reseller need no stock or post room (no inventory).

To create an online portal to connect makers and sellers, with a super-streamlined and simple workflow that connects both sides’ needs with each other, a very simple and intuitive UI capabilities with data capture functionality to create a marketing/customer profile to use to improve services to both types of user, e.g. what the sellers are searching for to enable the makers to improve their business plan/strategy.

1. We need a database that is suitable for both sets of users – makers and sellers.
2. The functionality needs to be simple and streamlined with the capacity to grow and be futureproofed, that captures all necessary information without ‘doubling up’ or capturing too much – most valuable data with fewest fields that enables future functionality and data capture.
3. To ensure energy sustainability of data storage.
4. The most workable/simple workflow for now and the capacity to change with any external changes.
5. Multi-platform app that would have the same ease/functionality/capabilities as the desktop database.
6. Inventory control for makers; tracking control for resellers
7. Payments portal/subscription/sign up portal
8. Good payer/other ratings capacity to enable a standard to achieve user protection
9. Ethical and standard rating - in line with sustainable development goals.

**Challenges presented by the project**

The project combines technical challenges and requires interacting with a client and working towards real and potentially changeable requirements.

**Learning outcomes**

This is software development in a real environment of a social enterprise startup. you will learn to: interpret user requirements; present design ideas in a technically accurate way and in a way which is comprehensible by a non-technical audience; plan project tasks to deliver defined targets within a prescribed time-scale.

**Nature of end-product**

Software and designs, documentation

**Project timetable**

The project will follow an agile approach, completing a cycle of requirements, design, prototype development, and feedback each month.

## Placements Management Web App

### Supervisor

J.M.Rojas (jmrs3)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Software Engineering for Financial Services (SFS)

Web Applications and Services (WAS)

Advanced Distributed Systems (ADS)

Advanced Software Engineering (ASE)

Cloud Computing (SCL)

Human Technology Interaction (HTI)

Advanced Computer Science (ACS)

### Prerequisites

Web development stack (Javascript, Java, REST, Spring)

### Aims of project

The Year in Industry, also known as Industrial Placement, is a core element in the course of many students at the School of Informatics. Students spend their third year gaining valuable working experience at a company of their choice in their topics of interest. While this happens, the School of Informatics, and in particular its Placements Tutor, must keep close contact with these students. At the moment, the management of this task relies on time-consuming and error-prone Word documents, Excel spreadsheets and emails. The aim of the project is to develop a web application to support the different stakeholders involved in a Year in Industry / Placement, including the student when submitting a placement authorisation request and the placement tutor when tracking students on placements.

### Challenges presented by the project

There are multiple challenges involved in this project. A reasonable selection of features may include:

- Allow students to submit authorisation requests. These authorisation requests must be confirmed by the placement provider and must afterwards be approved by a placement tutor.

- Manage students on placements: add placements, update placement (same company, e.g., change placement end date), change placements (student moves to a different company), etc

- Contact students on placements

- Schedule placement visits; involves integration with a calendar solution to schedule physical or virtual meetings between placement tutor, employer and student

- Map view and smart visit plan; Draw a map with pins on all the companies where students are currently on placement. If there are different students working in a certain area (e.g., South Yorkshire) suggest multiple visits at the same time.

- Allow students to contact the placement tutor, as an alternative to email communication.

- Allow students to submit placement reports (possibly integrating with Blackboard)

### Learning outcomes

The student will strengthen their current web development skills and develop new ones.

The student will learn how to develop a web-based solution to a challenging, realistic problem.

The student will learn how to integrate external tools and apis into their own software solution.

### Nature of end-product

A web-based platform to support placement tutor and students on placements.

### Project timetable

The solution will be hosted on the School's website and will be accessible for students starting their placements in September 2021.

## Online video platform

### Supervisor

P. Severi (pgs11)

### MSc Programmes

Web Applications and Services (WAS)

Advanced Computation Methods (ACM)

Advanced Software Engineering (ASE)

Advanced Computer Science (ACS)

### Prerequisites

Web application development skills

### Aims of project

An online platform where users can share videos.

### Challenges presented by the project

How to securely store and share data  amongst many users

Provide advanced search for videos

### Learning outcomes

How to design databases and code in the chosen framework and programming language.

### Nature of end-product

An online video platform

## Online Collaborative Recipe Book

### Supervisor

P. Severi (pgs11)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Web Applications and Services (WAS)

Advanced Computation Methods (ACM)

Advanced Software Engineering (ASE)

Advanced Computer Science (ACS)

### Prerequisites

Web application development skills.

### Aims of project

Develop a  web application where users can add, modifiy, rate,search and share recipes with other users.

### Challenges presented by the project

Recommendation of recipes based on past searches and profile of users.

Advanced search  which allow to include different fields and exclude others,

such "search for a vegetarian  desert which  contains cream  but no nuts"

or "search for  gluten-free starters containing chicken".

Find as many ways as possible to classify  recipes  to improve the search experience.  For instance,

- vegetarian, vegan  and non-vegetarian

- soup, desert ,starters, main dish

-   diary-free, gluten-free

### Learning outcomes

Learning how to build web applications.

Learning how to classify data in  many different ways either using a  database.

Learning recommendar systems to recommend recipes.

### Nature of end-product

A web application where users can add, rate,  search  recipes and  get recommendations based on their preferences.

## An automated system for local GPs

### Supervisor

N.Yap (ny50)

### MSc Programmes

Agile Software Engineering Techniques (SET)

Software Engineering for Financial Services (SFS)

Web Applications and Services (WAS)

Advanced Computation Methods (ACM)

Advanced Distributed Systems (ADS)

Advanced Software Engineering (ASE)

Cloud Computing (SCL)

Advanced Computer Science (ACS)

### Prerequisites

Any knowledge e.g. web technologies and database for building a web-based system.

### Aims of project

The project aims to deliver a secured online system for the surgeries.

The system should allow patients to check the availability of doctors/nurses and book the appointment online. In addition to that, the patients should be able to manage their medical prescriptions. For examples, buy and pay the prescriptions online, check if their prescriptions have arrived etc. The doctors/nurses should be able to insert/update their availability on the system.

### Challenges presented by the project

Delivering a complex and user friendly web-based system.

### Nature of end-product

Web-based application for the local GPs.