DANDELION DATA MANAGEMENT PLATFORM

DEVELOPMENT PROPOSAL

BRIAN DAVISON
EDINBURGH NAPIER UNIVERSITY
15 OCT 2021

Executive summary

The Computational Sustainability Lab at Edinburgh Napier University is pleased to present this proposal for a data collection and management platform in response to the call by the Dandelion initiative. The recommended solution envisages

- Data storage based on a relational database and the standard system file hierarchy
- Server-side components based on Python Flask implementing a REST API
- Front-end functionality based on the React JavaScript framework
- IoT data collection nodes based on the Raspberry Pi Zero W

The start date is 15 Nov 2021 with all major development complete by 14 Mar 2022 ready for an official launch on 18 Apr 2022. The total cost of the development is £48,850

Table of Contents

1. High-level requirements

The Dandelion project will involve around 500 schools across Scotland in activities related to the cultivation of plants. 100 participating secondary schools will be supplied with two Liberty Produce GrowCubes each for the purposes of conducting experiments on the effects of various factors on plant growth. The schools project will run throughout 2022, but the majority of activity will take place during the growing season between April and September.

A web-based technical solution is required which supports

- The publication of textual and visual content
- The management of novel projects based on the Liberty Produce growing cubes
- The collection of attitudinal data from participating schools
- Managed access to the various categories of data by appropriate users and systems

1.1. Scope and assumptions

This proposal includes the design and development of the technical systems required for the solution, the management of the technical work and the provision of documentation and support. It assumes that

- All technologies will be open-source and incur no licensing costs
- The system will be hosted on SRUC servers and there will therefore be no hosting costs
- Data collection nodes installed into the GrowCubes will be allowed to connect to the host school's WiFi network and transmit their data to a server across the Internet, but will be prevented from receiving incoming connections by the school's network infrastructure.
- A small number of schools will be identified as early adopters where staff will be available during the project for requirements analysis, feedback on interim prototypes and beta testing.

2. Technical solution outline

To avoid uncertainty that might lead to development delays, the solution will be based on tried and tested technologies. Given the short timescale, it will also be an advantage to break the work down in a modular fashion so that different parts can be progressed in parallel. This is already indicated by the series of target dates in the call for proposals and the main strategy proposed to achieve this separation is the use of a REST API. Figure 1 summarises the system architecture.

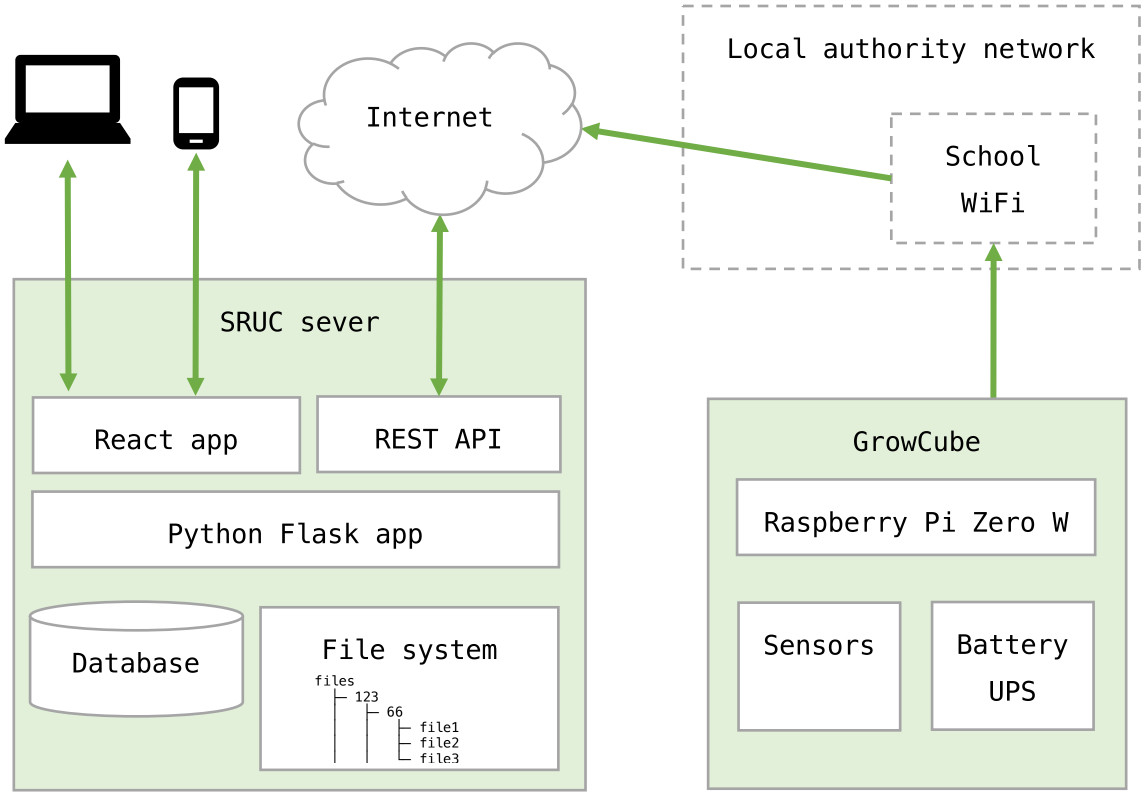

Figure 1: Proposed architecture

2.1. Structured data storage

A traditional relational database is selected as the main storage element for structured data such as numbers, text, dates and times since is more suited to use with a REST API than a document database such as MongoDB. The specific platform selected is MariaDB which is a 2009 fork of MySQL. MariaDB integrates with application code in an identical way to MySQL but is preferred because of its improved performance.

2.2. Unstructured data storage

While most relational platforms now also provide for the storage of binary large objects (BLOBs) which include images, video and audio, they may adversely affect performance.

The solution proposed here therefore uses the traditional approach of storing BLOBs as files on the server in a structured directory hierarchy. The file path to a specific object is stored in the database so that it can be easily retrieved. This approach takes advantage of the best features of the database and the file system.

2.3. Data ownership

Schools will be represented as records in a SCHOOL table in the database, each with a unique identifier. This identifier will be associated with all related records in other tables so that data ownership is transparently represented. For BLOB data, the school identifier will be used as part of the directory structure such that all material uploaded by a school is contained in its own subdirectory.

2.4. Data access

Access to all data in the system will be mediated either by the REST API directly, or the React app which provides a responsive interface for desktop and mobile clients. The system will implement role-based security in order to manage appropriate levels of access for different users.

2.5. Representation of experiments

Schools will be encouraged to define and run their own experiments. To provide the required degree of flexibility, an abstract project template that a school can configure when creating a new experiment.

A project is defined by a set of high-level data such as a name and description, and may include a series of experiments. An experiment may contain the following features:

- Variables (Required): Factors that can be either controlled or measured
- Observations (Required): Recorded variable measurements
- Hypotheses (Optional): Formal expectations on the experimental outcome
- Conditions (Optional): Contrasting combinations of variables in a single experiment

While many projects will be based in a single school, the system will also allow multiple schools to collaborate on a single project. In such cases, there will be a lead school responsible for maintaining the project information and the definition of experiments, and other schools will be able to contribute observations.

2.6. API

The API will conform to the REST OpenAPI 3.0 standard and will be designed using the online Swagger tools. This ensures that the API will be fully documented and that it can be fully tested in isolation. The API will be central in driving the development of the other main system elements and will therefore be completed early in the project.

Access to the API will be controlled through the use of JavaScript web tokens (JWT) following successful authentication using standard usernames and passwords.

Some schools may wish to access data directly through the API rather than through the React app. The API will be configured to allow read-only access to data by this method to avoid data errors, and documentation will be provided to support any schools that wish to take advantage of this option.

2.7. Web app

The technology selected for the user-facing web interface is the React framework. React provides many features which enhance the user experience including the ability to manipulate data in the browser and avoid full-screen refreshes. In addition, the ready availability of development tools speeds up the development activity and improves the consistency of the design through re-usable components. The presentation of the app will make use of the Bootstrap CSS framework and the Material Design graphical language to ensure that the user interface works well on bot desktop and mobile devices.

During the design phase, the user interface will be elaborated as a low-fidelity prototype in advance of any software development. This activity helps to identify and resolve and issues to do with functionality and navigation without the need to repeat development work.

2.8. Data exploration dashboard

Different user roles will be presented with variations of the main dashboard to reflect their specific requirements, For public users, the dashboard will be geared towards exploring the data. There are two main steps in this process:

- Data selection
- Data visualisation

The public dashboard will present a range of methods for exploring the available datasets through keyword search and browsing/filtering functionality. This will include the ability to combine compatible datasets – for example, from different schools – into a single data source. Compatibility will depend on the way in which the dataset has been defined. Having selected the data to visualise, the dashboard will allow the user to select from a menu of visualisation types that are appropriate to the selection. A time series, for example, might be best represented by a line graph while a pie chart might be more appropriate for summary data on survival rates. Visualisations will be generated using chart.js which integrates well with React and provides a comprehensive range of chart options.

One of the central visualisation methods available will make use of data overlaid on a map. This will allow for country-wide overviews of the data collected as well as comparisons between regions, local authorities and individual schools. This type of presentation will offer the option to drill down to the raw data which can then be exported for further analysis.

To encourage a degree of healthy competition, the dashboard will also allow the user to compare schools on certain overall statistics such as the number of experiments being undertaken, the quantity of raw data uploaded, the number of registered accounts, etc.

As part of the overall Dandelion initiative, anonymous attitudinal data will be collected from participants which can be visualised using the same process as for other data. For example, the user might wish to see the changes in responses to a particular question over time which would involve selecting that particular question and choosing a line chart visualisation.

The dashboard for other roles will incorporate features relevant to the role. For example, The main school account will have access to features for managing the additional accounts for the school. School staff and students will have the ability to define projects and experiments, and to upload data.

2.9. User guidance documentation

The month preceding the launch date is set aside for user testing. During this period, the documentation producing during development will be supplemented by how-to guides and video material in collaboration with the SRUC team.

Following the launch, technical support will be available from the project lead, and potentially from other member of the development team. At this point, the GitHub repository will be made available as open source according to FAIR principles.

2.10. IoT nodes

The budget for this proposal includes the purchase of 200 Raspberry Pi Zero W single-board computers that will be installed in the GrowCubes. These devices will act as Internet of Things (IoT) nodes, collecting data automatically via sensors and transmitting that data to the REST API for storage. Each node will include

- Raspberry Pi Zero W in a protective case
- Uninterruptible power supply (UPS) based on a lithium polymer (LiPo) battery
- Basic sensors: temperature, humidity, water level, light intensity, noise level

3. Previous experience

The project lead, Dr Brian Davison, has over 30 years' experience in software development both within the higher education sector and industry. The majority of his projects have employed web technologies. As part of his current role, he leads the undergraduate group project module at Edinburgh Napier University which runs on an annual cycle with between 40 and 50 teams running projects for live clients. Brian is also the lead for the University's Computational Sustainability Lab which is currently engaged in a similar project to collect data from a permaculture space and expose that data via web visualisations. Since 2013, Brian has led 17 successful consultancy projects with industry partners.

Links

- Brian Davison's University profile
- Computational Sustainability web page

4. Technical capacity

The proposed solution draws on technologies that are familiar to the team and to Edinburgh Napier's School of Computing in general. Two temporary dedicated developers would be employed for the duration of the project. One of these is a recent graduate from a Masters programme in web development. The second recently completed an undergraduate degree in computer science with a first class result and is now working towards establishing a start-up company specialising in web and mobile development.

Work on the data collection nodes integrated into the GrowCubes will be carried out by a current student as the main activity in his final-year Honours project. The student in question has previous experience in this area having carrier out similar work during his group project in third year. His results are now being further developed as part of the Smart Garden project.

Software development work will be carried out on the developers' own personal computers and deployed to the live server for integration testing purposes. It is assumed that the live server will be provided by SRUC. Alternative options are available if the live server is delayed for any reason. The first of these would be to create an integration environment on Edinburgh Napier's virtual infrastructure.

The team would be supported if necessary by other technical resources in the School of Computing and the wider University.

5. Health and safety

Edinburgh Napier University defines policies for working times and conditions that the project teams will adhere to. The majority of the work will consist of software development which carries little inherent risk. All members of the team are familiar with the need to control the physical working environment and to protect against issues such as repetitive strain injury. The majority of the development work will be done remotely, and the risks related to the transmission of COVID-19 will therefore be minimised.

Some visits to schools will be necessary as part of the development work, especially during the beta testing phase in March 2022. This will involve some travel and appropriate measures will be taken to ensure that this is done in as safe a manner as possible with respect to any health guidance in force at the time.

Some element of risk will be incurred through the installation of data collection nodes in the Liberty Produce GrowCubes. The proposed processing unit is a Raspberry Pi Zero W which operates at 5V. At this voltage, there is no risk of electric shock. The Raspberry Pi will however need to be connected to mains power for most of the time. In the case where the Raspberry Pi gets wet while plugged in, the on-board safety fuse will prevent any unwanted over-current from the mains. The only damage in such circumstances is the need to replace the Raspberry Pi itself.

A further risk regarding the Raspberry Pi is the inclusion of a lithium polymer (LiPo) battery to allow the unit to be disconnected from the mains supply for short periods while continuing to operate. LiPo batteries can get hot if short-circuited which could be the case if the unit were to get very wet. Several steps will be taken to mitigate this risk as described in the list below.

- The Raspberry Pi will be installed on the outside of the GrowCube away from the area where water is in use
- The Raspberry Pi will be housed in a protective case
- Guidance and operating instructions will be provided for school staff and students
- The health of each Raspberry Pi will be monitored by the system and an appropriate person will be alerted of any unexpected events

6. Security

An explicit requirement in the brief is that no personal data should be stored in the system. Users will therefore be represented only by ID numbers. There will be no self-registration feature in the system; instead, a main school account will be created by the overall system administrator. The school account holder will have the ability to create accounts for additional users in the same school. The additional users might be staff or students, and it would be the school's responsibility to track them. Each account would be protected by a strong password, and the main school account holder would be able to reset passwords for the additional accounts. Further protections on account access could be explored if this is deemed necessary.

It is assumed that the data collection nodes in the GrowCubes will transmit data to the central server, but will be prevented from receiving incoming connections by the school's firewall router. This will eliminate the possibility of unauthorised intrusions. All communications with the server – including from the data collection nodes – will be secured using HTTPS.

Operating system access to the stored data on the server will be permitted to a small controlled number of user accounts. Otherwise, the data will only be accessible via controlled interfaces such as a school's management dashboard and the main API.

7. Quality and timescale

Given the short timescale, a hybrid approach to project management will be adopted in which three main phases are identified. The second and third of these will be based on iterative agile methods. A phased structure allows for the elaboration of requirements at the start of the project including the definition of acceptance criteria, while an agile approach in the later phases give the team the ability to react quickly to feedback as the work progresses. The detail of the project plan can be found online and the main phases are as follows.

- 15 Nov 2021 – 10 Dec 2021 Design phase
- 13 Dec 2021 – 4 Mar 2022 Development phase
- 7 Mar 2022 – 18 Apr 2022 User testing phase

The main quality safeguards for the project overall are:

- The regular inclusion of early adopter school staff in all phases
- The agreement of acceptance criteria during the design phase
- The active use of MoSCoW task prioritisation
- The adoption of a defined GitHub workflow for all project tasks
- The explicit definition of ready (DoR) and definition of done (DoD) for all tasks

During development, GitHub tools will be used to ensure that all activity is visible to the project team and to any interested stakeholders. This will include

- Quality criteria for code check-in comments
- Project status tracking using a Kanban task board
- Peer reviews of code before merge
- Use of a feature-based branching strategy

8. Costs

The costs for this project have been calculated using the standard consultancy tables for Edinburgh Napier University. There are three main elements as shown below.

Item	*Cost*
Staff costs (including overheads)	£36, 490
Equipment (IoT node materials)	£12, 000
Travel and subsistence (for school visits)	£360
Total	$48, 850