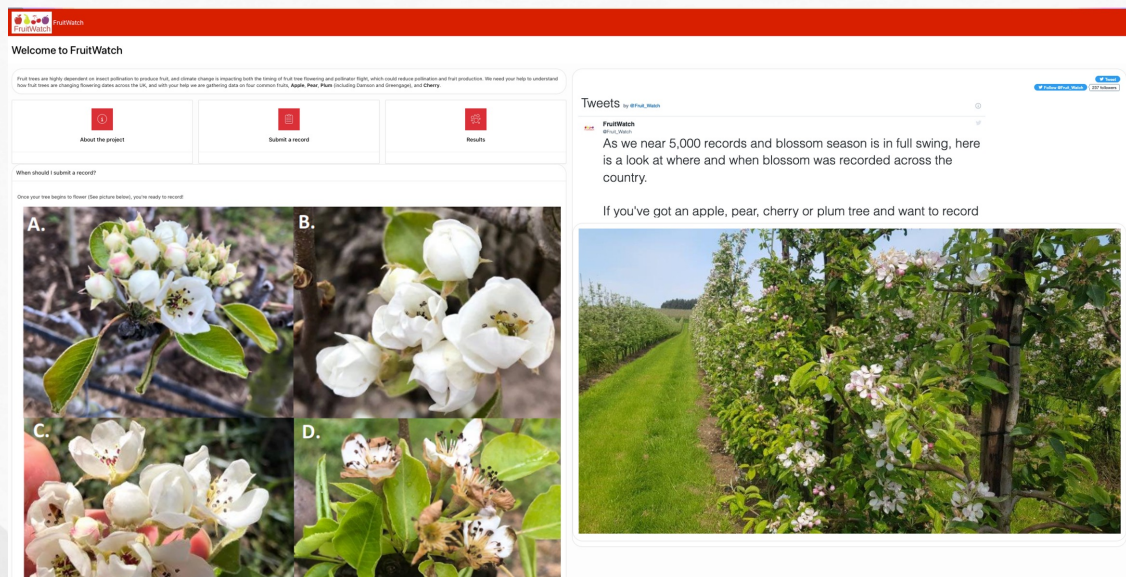


Technical Brief

University of Reading enables citizen science with Oracle Autonomous Database and APEX

Here we explain the technical approach adopted for one of our projects and its requirements to capture data. The researchers wanted to engage the public to collect data on when their fruit trees flower. Thus the 'citizen science' Fruitwatch.org program was conceived – getting the public to record data from the whole of the UK that can then be analysed with other data – a key component in this research project. It would be impossible to obtain this data in any other way.

Within this technical brief, we delve a little deeper into the technology and technical configuration that has rapidly and easily supported the science.



University of Reading launches citizen science project Fruitwatch.org

The study aims to monitor changes and trends in fruit tree flowering dates across the UK. Fruit trees are highly dependent on insect pollination to produce fruit, and climate change is impacting both the timing of fruit tree flowering and pollinator flight, which could reduce pollination and fruit production.

We hope the findings of this project will enhance our understanding in variation of fruit tree flowering across the UK. It is part of a PhD project entitled “Mitigating the risks to pollination services caused by climate change” at the University of Reading. It is a project that is helped by an Oracle for Research grant which includes OCI credits and support. Citizen science enables the data to be captured. The launch of [Fruitwatch](#), along with its intentions and goals are described here within the [OfR Fruitwatch launch blog](#).

Project Requirements

The requirements were straightforward: -

- To design and implement a system that is accessible to all so that citizen scientists can record fruit tree flowering information.
- The system will be owned, implemented, and maintained within the University of Reading OCI tenancy provided by the Oracle for Research project award.
- The task of collecting the data should be simple and easy to perform. No personal information will be gathered, just key information about tree, flowering state, the date & location as indicated by browser location or dropping a pin on a map or entering a valid postcode.

Allow images of flowering trees to also be uploaded to enhance the data and analysis. The system should be accessible to all – browser based – PC, Laptop, or mobile device.

How to quickly and easily write a cloud application



[Autonomous Database](#) is the perfect fit for this. It provides a secure, robust data platform in which the data can be stored, managed, and used. It requires no database management – patching, backups, scalability, and availability are all provided for you - so our researchers can concentrate and focus on the data and the science; its ideal.



In addition, it is so easy and quick to write an application that can be used to capture the required data from the public. [APEX](#) is the low code application tool that comes for free with your Autonomous Database. This was used to develop the application.

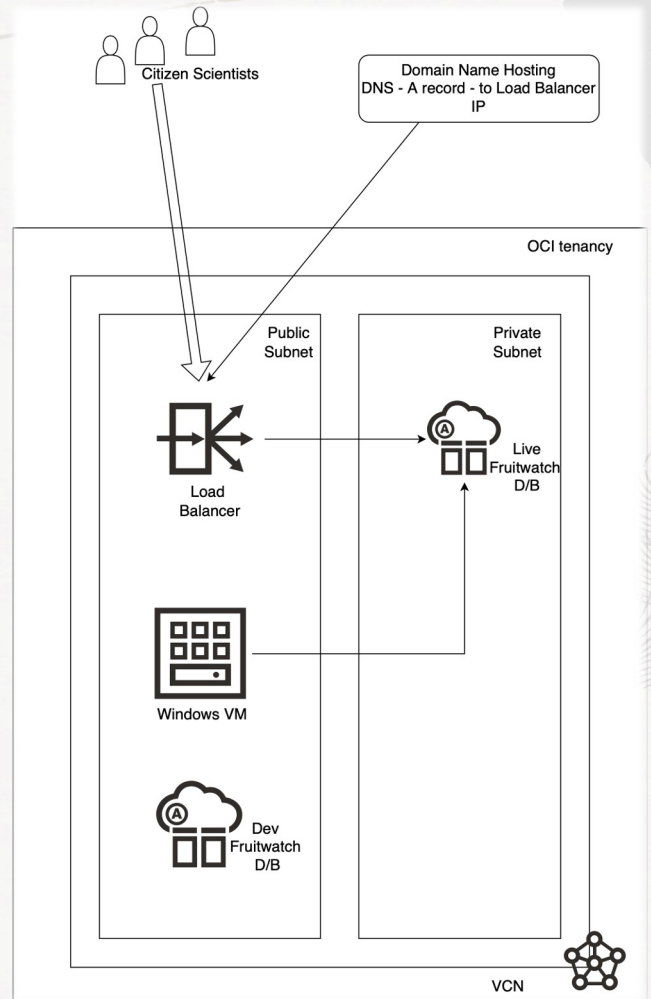
With support from Oracle for Research technicians a significant part of the application was ‘written’ by the researcher Chris Wyver from Reading Uni – this was his first introduction to APEX.

What's the solution architecture?

The implementation is best seen from the architecture diagram.

The OCI services in use are:

- Oracle Autonomous Database
- Oracle Analytics
- Compute
- Storage



There are a couple of areas to note: -

1. The development environment APEX, ADB tools were accessible from the internet – this enabled easier collaboration and access for all team members during the development phase.
2. The production database environment was placed into a private subnet. To provide a simple way to get to the application (Fruitwatch.org) the following blogs series (first 3 blogs) were used to ensure that we provided a simple URL for our citizen scientists to use:

<https://blogs.oracle.com/apex/post/introducing-vanity-urls-on-adb>

The only additional configuration that was carried out in addition to these was to redirect port 80 to 443 – all access is via https. This was configured within a rule set within the load balancer.

Edit Rule Set Help

Specify the rules that control traffic flow through the listener.

Name
DefaultRedirect

Specify Access Control Rules
 Specify Access Method Rules
 Specify URL Redirect Rules

URL Redirect Rules

URL redirect rules specify how to route incoming HTTP requests to a different destination URL.

Source Path Match Type

Redirect to: ⓘ Switch to full URL

Protocol <input type="text" value="HTTPS"/>	Host <input style="float: right;" type="text" value="{host}"/>	Port <input type="text" value="443"/>
<small>Original value: (protocol)</small>	<small>Original value: (host)</small>	<small>Original value: (port)</small>
Path <input type="text" value="/{path}"/>	Query <input style="float: right;" type="text" value="{query}"/>	Response Code <input type="text" value="302 - Found"/>
<small>Original value: (path)</small>	<small>Original value: (query)</small>	

Specify Request Header Rules
 Specify Response Header Rules
 Specify HTTP Header Rules

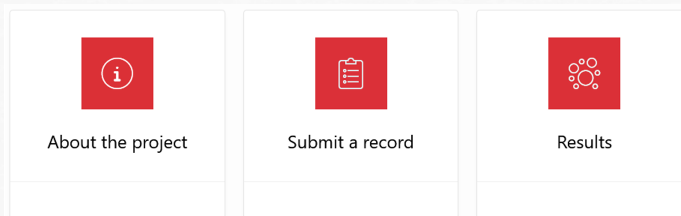
3. For direct access to the production APEX and the ADB environment within the private endpoint the Windows VM is used to provide direct browser access.

Using Fruitwatch.org, an APEX application

Citizen scientists can access the application at [FruitWatch.org](https://fruitwatch.org). The application guides the user through a couple of simple forms to allow them to enter information about the fruit trees, location stage of flowering and the date. It also allows for images of the flowers to be uploaded.

Careful attention was given for instructional text and steps on how to use the application, along with a Privacy Statement & Terms of Use

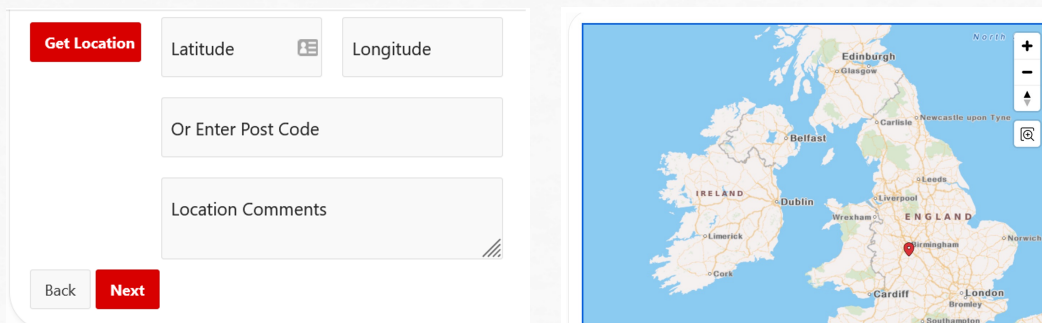
A report region is used to display Tweets related to the project's Twitter account, along with sample images throughout of different flowering stages for context and reference.



Cards: <https://blogs.oracle.com/apex/post/a-simple-guide-to-the-new-cards-region-in-apex-202>

Cards is a native APEX report region type, which was implemented to present a simple navigation UI and functions in the application. Very little customization was needed, apart from settings for color and icons

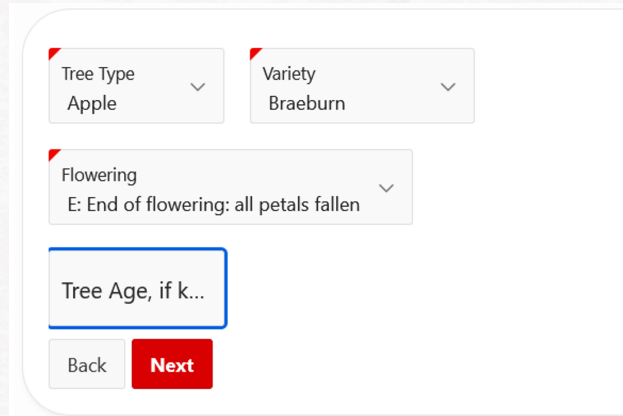
Step 1: Setting a location on a new record

The image shows a user interface for selecting a location. On the left is a form with a red 'Get Location' button at the top left. Below it are two input fields: 'Latitude' with a small icon of a coordinate system, and 'Longitude'. Below these is a text input field labeled 'Or Enter Post Code'. At the bottom of the form is a larger text area labeled 'Location Comments'. At the very bottom of the form are two buttons: 'Back' and 'Next'. To the right of the form is a map of the United Kingdom and Ireland. The map shows major cities like Edinburgh, Glasgow, Belfast, Dublin, London, and Cardiff. A red pin is placed on the map, indicating a selected location in the Midlands region of England.

Location is gathered with different methods:

- A plugin is used to get location information from the browser, whose browser execution must be authorized by the user
- Drop a pin for approximate location on the Map region - <https://apex.oracle.com/en/platform/features/whats-new-211/>
- Post Code - PL/SQL Validation of correct alphanumeric string format for UK post codes. And since the mapping is based on lat/long coordinates, if the user does not enter the information, it is looked up from a list of valid post codes.

Step 2: Capturing tree type and flowering records

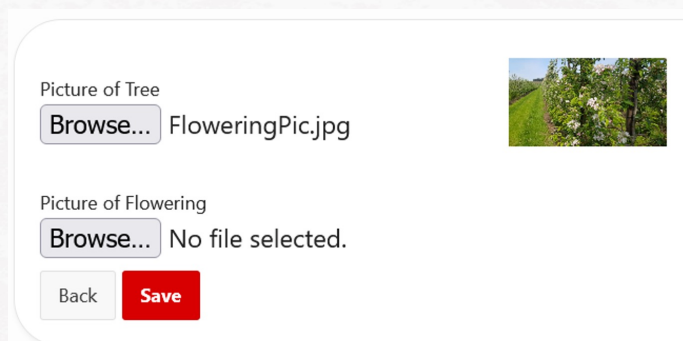


The screenshot shows a data entry form with the following fields and controls:

- Tree Type**: A dropdown menu with "Apple" selected.
- Variety**: A dropdown menu with "Braeburn" selected.
- Flowering**: A dropdown menu with "E: End of flowering: all petals fallen" selected.
- Tree Age, if k...**: A text input field with a blue border, currently empty.
- Navigation**: "Back" and "Next" buttons.

The application allows the user to select Tree Type, it's related Variety (if known, or can enter text for a Not Listed variety), Flowering Stage, and approximate Tree age.

Step 3: Submitting pictures to track flowering stage



The screenshot shows an image submission form with the following elements:

- Picture of Tree**: A "Browse..." button followed by the filename "FloweringPic.jpg" and a thumbnail image of a tree in bloom.
- Picture of Flowering**: A "Browse..." button followed by the text "No file selected."
- Navigation**: "Back" and "Save" buttons.

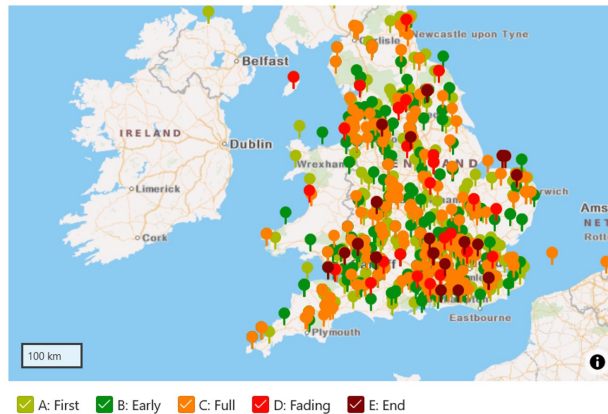
This allows the user to submit 2 image files, which can use the native camera application or photo library functionality from a mobile device. An APEX File Browse page item is used, with browser-native formatting. Once selected, a thumbnail representation of the image is shown.

And that's it! Once submitted, the user will be shown the Results page, which includes a Record Summary of the records submitted for each Tree Type. Maps are also shown on the Results page, one for each Tree Type, with different layers indicated by pin colors for the different flowering stages or each.

Results

Record Summary

Fruit Type	Number of Records
Apple	1590
Cherry	872
Pear	916
Plum	2509



The future of the project

The data capture is happening now – a big thank you to all those people that have connected to Fruitwatch.org to record the information. This is just the beginning though – the data from the public will be invaluable in being analysed with other data e.g., weather data, field data to be able to answer the scientific questions. The data can all be stored, manipulated, queried, and reported on, all from the same Autonomous Database – all the tools and capabilities are provided to be able to utilise the data, APEX, SQL, machine learning, Big Data.

APEX and ADB are provided within the [Oracle Cloud Free Tier](#).

They are also accessible by applying for one of our awards at [Oracle for Research](#)

The background features a light gray color with abstract, artistic elements. There are several circular fingerprint-like patterns in the corners, some in white and some in a darker gray. Additionally, there are several long, thin, curved brush strokes in white and dark gray, creating a sense of movement and depth.

oracle.com/research