Health Research from Home (HRfH)

Hackathon 2025

Participant Manual and Guidelines

Introduction

Welcome to the Health Research from Home (HRfH) Hackathon 2025, hosted by the University of Manchester. This hackathon aims to bring together early career researchers and public contributors to collaboratively explore innovative solutions and practical tasks in health research using electronic patient-generated health data (ePGHD). The event emphasizes building a collaborative community that enables participants to engage deeply with real-world health challenges through multidisciplinary teamwork, fostering connections, and creative problem-solving. This manual serves as your guide, providing essential information on navigating the event, understanding your tasks, accessing resources, and maximizing collaborative outcomes. We look forward to seeing you all in Manchester!

Contact us: hrfh@manchester.ac.uk

Teams channel: Teams

Teamwork folder: SharePoint Folder

Project submission folder: Submission

WiFi access: WiFi Info

Event Schedule

The hackathon will be held at the Christabel Pankhurst Institute (located on Dover Street, M13 9PS), University of Manchester, from May 7 to May 9, 2025.

Wednesday 7th May	
Registration & breakfast	9:30 – 10:00
Introductory talks: • Introduction to Health Research from Home – Will Dixon • Introduction to Instructors & Facilitators – Will Dixon • Background to Task 1 and Task 1 definition – Will Dixon, Catherine Irwin & Paul Amlani-Hatcher, Mariam Al-Attar	10:00 – 10:45
Ice breaker, introduction to delegates & team formation	10:45 – 11:30
Groupwork	11:30 – 12:30
Lunch (provided)	12:30 – 13:30
Group work	13:30 – 16:30
Mo Malekzadeh - 'Multimodal Wearable Sensing for Preventive Healthcare'	16:30 – 16:45
Day 1 - Q&A session	16:45 – 17:00
Evening meal (provided) • Location: Bundobust at St James Building, 61-69 Oxford Street, Manchester, M1 6EQ	19:00 – 21:00
Thursday 8th May	
Breakfast (provided)	09:30 – 10:00
Task 2 definition	10:00 – 10:30
Dr. Matt Sperrin - 'Adaptive sampling: deciding what to collect and when.'	10:30 – 11:00
Group work	11:00 – 12:00
Lunch (provided)	12:00 – 13:00
Group work	13:00 – 16:00
EOD meeting, troubleshooting & planning for day 3	16:00 – 17:00
Friday 9th May	
Breakfast (provided)	09:00 – 09:30
Group work	09:30 – 11:00
Submission deadline	11:00
Live demos	11:00 – 12:00
Lunch (provided)	12:00 – 13:00
Christopher Yau - Talk title TBC	13:00 – 14:00
Group presentations on Task 1	14:00 – 15:00
Results & Group feedback	15:00 – 16:00

Task Descriptions

Task 1: Simulation of Tracked Daily Symptoms

In this challenge, you will build a synthetic data generator that simulates daily pain measurements for a group of imaginary people living with rheumatoid arthritis over a six-month period. Unlike a standard cross-sectional dataset, this simulation should reflect longitudinal patterns that evolve over time and capture dynamics of disease progression and treatment response.

Clinical and patient partners will share key insights during the hackathon to guide how these patterns could be represented. Your final output should be a realistic synthetic dataset that can be used in subsequent modelling tasks.

Your Challenge

Design a clinically realistic simulation capturing the following features:

1. **Changing disease state:** Simulate realistic long-term pain patterns, including gradual deterioration and periods of improvement
2. **Daily variability:** Include random (but plausible) daily fluctuations in pain scores to reflect natural day-to-day variations. Ensure diversity among patients so no two individuals would have identical pain trajectories.
3. **Flares:** Incorporate periods where symptoms suddenly get worse
4. **Treatment effects:** Incorporate at least one treatment event per patient, showing effects such as reduced pain after medication initiation, and varying extents and durations of treatment effectiveness.

You should document any key assumptions you make when creating the data generator, and make sure the data you generate is structured clearly for any downstream data analysis tasks.

As an example, a typical generated pain dataset might include columns such as patient ID, date/time of observation, a pain score on a fixed scale (0 [no pain] to 10 [worst possible pain]), and whether or not an intervention had taken place.

Although your simulator will be developed for RA, please include in your presentation how it would be adaptable for other long-term conditions. Please also show how a future researcher would be able to use your simulator – ideally having some control over disease patterns and treatment effects.

Instructions

- Your task is to create a re-usable simulator of daily tracked pain symptoms for people living with rheumatoid arthritis. Imagine the simulator will be used by future population health researchers who want to create a new simulated

dataset incorporating their pre-defined features in order to develop and evaluate new methods (e.g. ability to detect number or severity of flares).
- Generate daily pain scores for 5 example synthetic patients (see Visualization below). Pain should be represented on an ordinal scale from 0 (no pain) to 10 (extreme pain). Patients are assumed to be monitored daily for a 180-day period (i.e. from day 1 and day 180 with no break in-between).
- Consider how future users will be able to understand and make use of your simulator. This should include understandable code and ease of use.

Visualization

To verify and explain your simulation, visualize the full course of 180 days for a subset of your simulated patients:

- Use clear plots, with days on the x-axis and pain scores on the y-axis.
- Mark the timing of treatment events clearly on your visualization, for example, with shaded regions marking time periods when patients received treatment.
- Include five illustrative example trajectories from different patients to illustrate diversity.

Reproducibility and interpretability: Your method should be implemented in an open-source programming language (we recommend R and/or Python) standard script (.R or .py file) that can be executed from either a programming IDE or command line interface. All scripts contributing to your method must be contained in a single directory. The directory should be organised; scripts should be clearly commented and a readme.md should be included to explain how to run your generator on a new computer machine. This may also take the form of a notebook (e.g., R Markdown.Rmd, Quarto.qmd, Jupyter.ipynb) that renders from start to finish without user interaction or unspecified external dependencies (no disordered code chunks!).

Expected Output Format

Submit at **11:00 on Day 3 (May 9th)**:

- Simulator code (clearly commented and reproducible).
- Visualization plots demonstrating the first 180 days for five patients.
- Presentation slides explaining your simulation approach, assumptions, and key findings.

Clearly label your files and folders with your team's name for easy identification. Double-check submission completeness and readability before the final submission.

Oral presentation (10m) at **15:00 on Day 3 (May 9th)**:

- You have 10 minutes to present your simulator in the best possible light. The **evaluation criteria** are included below, so be sure to address those points.
- Consider including the following (although this is not prescriptive): Describe the development of your simulator. What did you learn from the talks and discussions with public contributors and clinicians? How did this influence the development and coding of your clinical features? How did you approach coding? Show how your simulator works. How did you consider re-use by future researchers? Show and describe the five outputs of your simulator.

Be Creative

Feel free to experiment with:

- Different modelling methods.
- Diverse patient scenarios and parameter variations (e.g., treatment timing, effect duration, intensity of fluctuations).

Aim for realism, creativity, and clarity.

Evaluation Criteria

Your simulation task will be evaluated based on:

- **Clinical plausibility:** Pain levels and trends must be clinically plausible, avoiding unrealistic jumps or perfect patterns. Each of the four features will be assessed independently
- **Adaptability and usability:** Clearly describe how the simulated clinical features can be adapted or extended in other contexts and to facilitate interpretation, reuse, and reproducibility by future researchers.
- **Presentation style:** How clearly you convey the development, function and output of your simulator, including clarity of presentation, audience interest, use of audiovisual support etc.

Have fun building your simulation!

Task 2: Predicting Pain Measurements and Optimizing Sampling Frequency

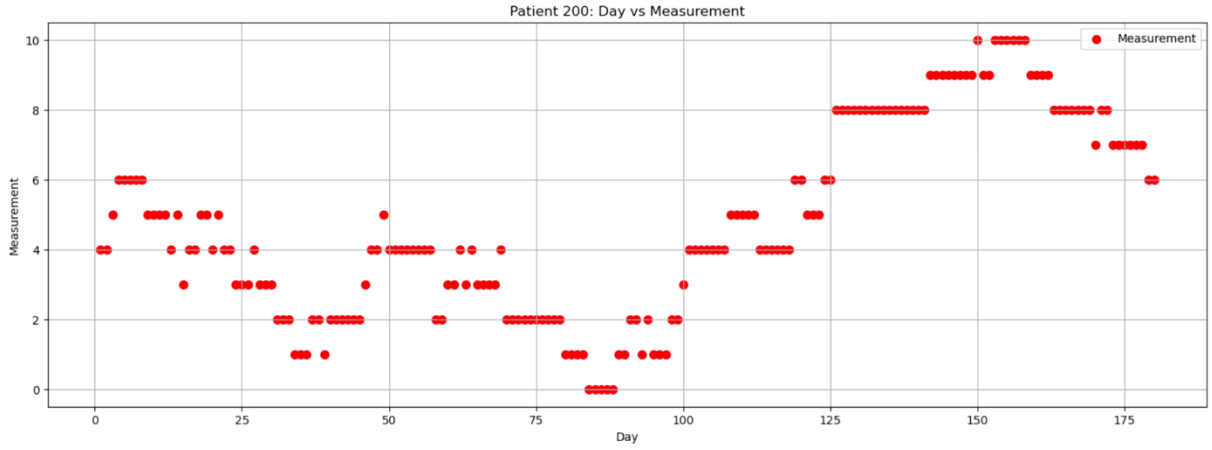
Overview

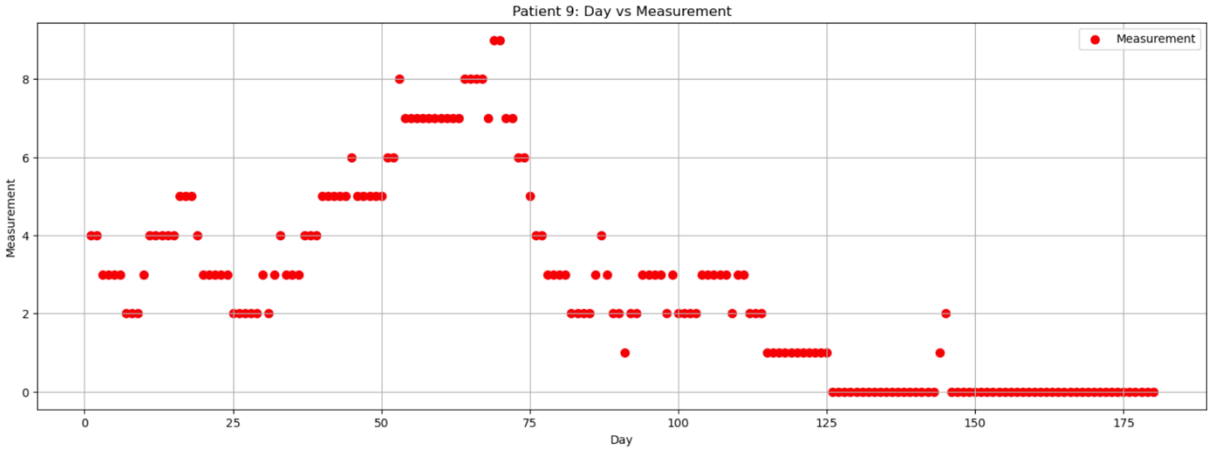
Imagine you have data on a number of patients, each with daily pain measurements recorded over a certain time period, and that you want to predict the future pain score trajectory of these patients. Ideally, using patient data from all available historical times would yield a more accurate prediction. However, in the real world, reporting outcomes can be time-consuming and burdensome to patients. Ideally, we would want to minimize this burden, requesting the minimum amount of data without losing important information. Your second task is to forecast pain reports of patients with high accuracy, while also minimizing the number of data points you have to retrieve in order to predict these scores.

Your Challenge

Data Description

You have access to daily, uninterrupted ordinal pain measurements for 800 virtual patients, each spanning a period of 180 days. The provided dataset includes complete daily pain data for every patient:



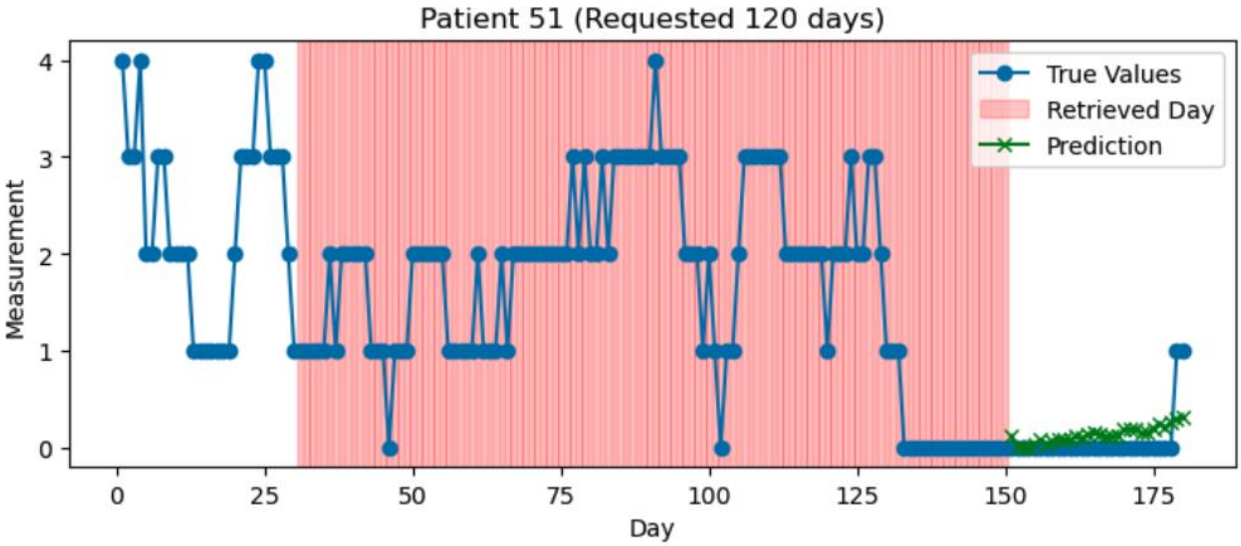


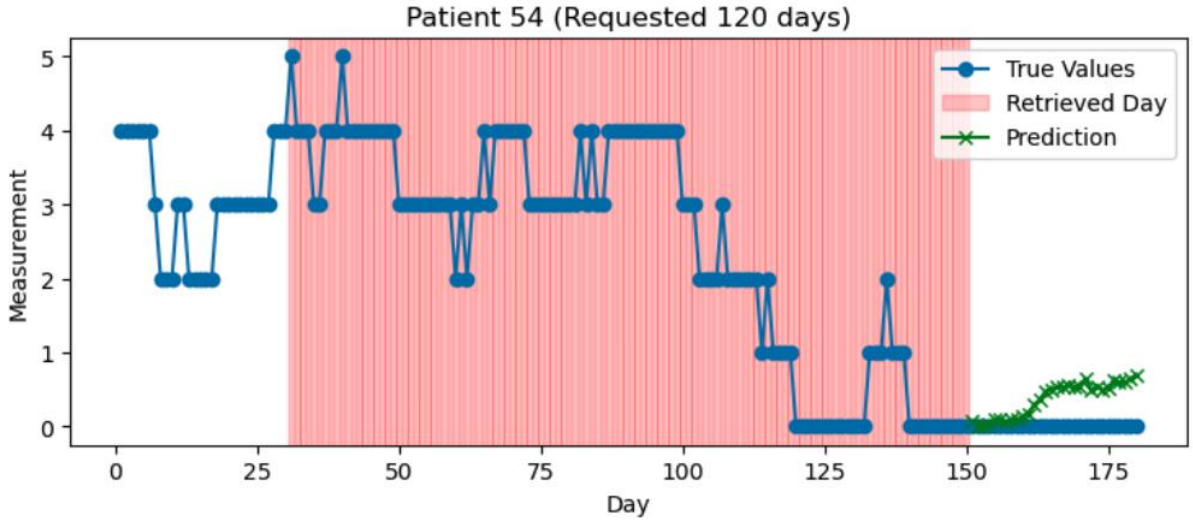
- **Days 1–30:** Initial baseline period (free and always available).
- **Days 31–150:** Optional measurement days (you can strategically choose which days to 'purchase' and measure symptoms – it is not necessary that you select consecutive days).While each day from day 31 to 150 will be retrieved at a cost, days that are closer to the end of this time period (i.e. days that are closer to the prediction period) will be more costly than those near the beginning.
- **Days 151–180:** Prediction period (you must forecast these pain levels).

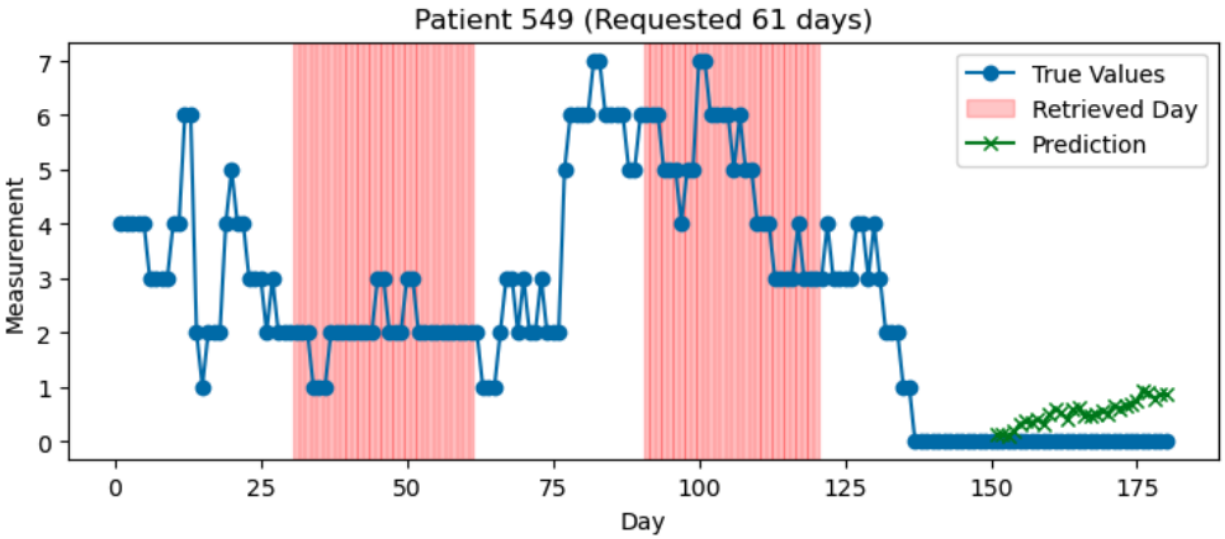
After developing your algorithm using this training dataset, **it will be evaluated on a separate, unseen test dataset.**

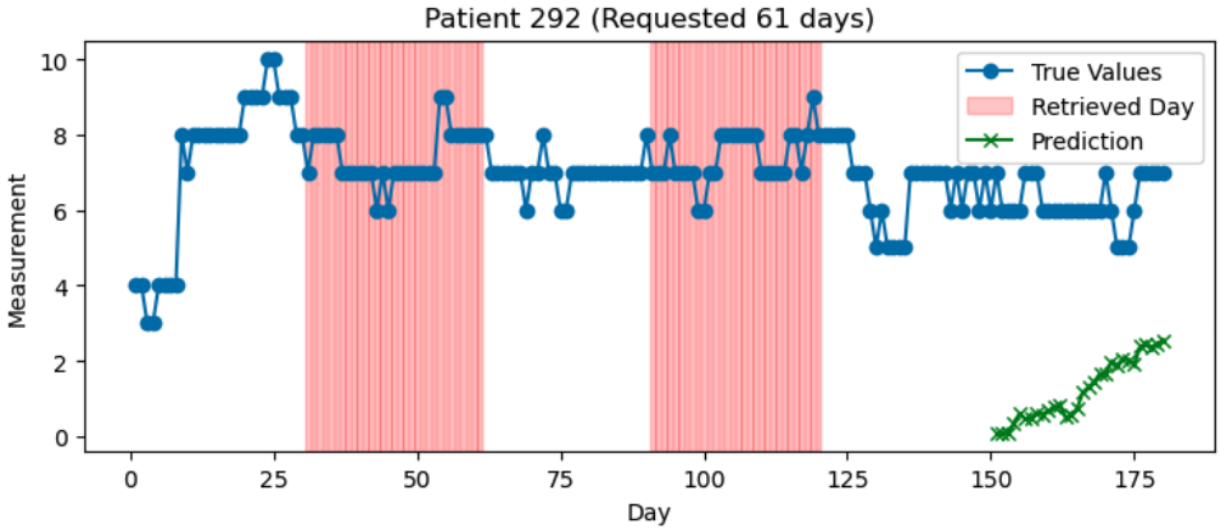
Develop an algorithm that:

- Selectively chooses a minimal yet effective number of days (between days 31–150) to measure each patient's symptoms. You are allowed to sample different days in different patients – see the visualizations below.









- Uses the selected measurements (plus baseline data from days 1–30) to accurately predict pain scores for the last 30 days (days 151–180).

Your goal is to achieve high prediction accuracy while minimizing data collection costs. Instructions:

- Use the provided training dataset (800 patients with complete 180-day measurements) to design and validate your prediction and sampling strategy.
- Your algorithm must include a strategy for selecting the optimal sampling days from days 31–150. Clearly document your approach for choosing these days (e.g., fixed intervals, adaptive approaches based on symptom trends, or machine-learning-driven methods).
- Balance carefully between reducing data collection (lower costs) and maintaining prediction accuracy.

Evaluation Criteria

Your submission will be scored on an unseen test dataset based on:

- **Prediction Accuracy:** Prediction accuracy measures how well your algorithm forecasts actual pain measurements during the final 30-day prediction period, assessed using **Quadratic Weighted Kappa (QWK)**.
- **Analytic Efficiency:** Running your code of entire process should not take more than 30 mins. Otherwise, 10% will be deducted from the final mark for Task 2.
- **Sampling Efficiency:** A penalty will be applied based on the proportion of data retrieved (days 31–150). A composite score combining QWK with data retrieval will be calculated:
  - $Score = QWK + \lambda \sum \frac{1}{151-d}$

    where λ is a penalty factor, and the sum is over all the days d sampled.

    The best submissions will have low prediction error with minimal data usage.

Expected Output Format

Submit at **11:00 on Day 3 (May 9th)**:

- Prediction: Please refer to Task 2 folder (Jupyter Notebook) for step-by-step instructions on accessing training/testing data and generating your final evaluation score for the hackathon. Make sure you can run the template successfully in advance—this is crucial to ensure your submission can be evaluated properly on Day 3 of the hackathon.
- Code: Clearly documented code used for training and generating predictions, using either R or Python. You should use the classes provided in code templates evaluation_functions.py or evaluation_functions.R in order to sample from the dataset. A guide in how to use this class is provided in notebooks task2_evaluation_template_python.html and task2_evaluation_template_R.html in the GitHub repository.
- ReadMe file: Explanation of your prediction method, data selection strategy, and instructions to replicate results.

Data and Resources

Datasets and detailed task instructions are accessible via the official GitHub repository: https://github.com/Health-Research-From-Home/DataAnalysisChallenge

The training data for Task 2 can be found here: https://github.com/Health-Research-From-Home/DataAnalysisChallenge/tree/main/Task%202

Submission Guidelines

- The deadline for both **Task 1** and **Task 2** is at **11:00 on Day 3 (May 9th)**
    - Submit your code here: Submission
- Code: Clearly documented R or Python scripts.
- ReadMe Documentation: Methodology, assumptions, and execution instructions.
- Visual Presentation for Task 1: Participants are required to prepare slides for a structured presentation of their Task 1 results. Our expert panel will evaluate and score your work based on presentations, clarity, and the realism of the simulated data. Limit your presentation to a maximum of 10mins.
- **Live Demo**: from **11:00 to 12:00 on Day 3 (May 9th)**
    - Task 1 – Simulator Demo: show how your simulator works by generating data or graphs of the same quality as those presented in your presentation.
    - Task 2 – Evaluation Demo: run the provided evaluation method to calculate the score for Task 2. Please ensure your code follows the output format and instructions provided notebook: Notebook.

Usage Rights for Submitted Project

All code and other materials created during the Hackathon event by you shall be referred to as your Results. By submitting your Results, you grant The University of Manchester ("University") an unrestricted worldwide, royalty-free, non-exclusive, irrevocable license to use, modify, distribute, and make your Results available on open-source platforms such as GitHub. This license includes the right for the University to use, adapt and share with third parties your Results for academic teaching and research purposes.

Contact and Support

- Event Leads: Prof. William Dixon & Dr. Shuai Shao
- Event Admin: Coral Stevenson
- Expert Panel: Prof. William Dixon, Dr. Glen Martin, Prof. Christopher Yau, Catherine Irwin, Paul Amlani-Hatcher, Dr. David Selby
- Facilitators: Dr. Mariam Al-Attar, Dr. David Selby, Dr. Shuai Shao, Dr. Emma Pritchard, Zhengmao Li, Jose Benitez-Aurioles, Aashna Uppal
- Public Contributors: Catherine Irwin, Paul Amlani-Hatcher
- Comms: Fu Lian Doble