Requirements Analysis Document

Sheep Methane Production Calculator Project

Group B

CITS3200 Professional Computing

Semester 2, 2011

University of Western Australia

Crawley, WA 6009

Revision History:

Version R0.1 22/08/2011 Andrew Cannon. Created from template.

Version R0.2 22/08/2011 Andrew Cannon. Revised following group requirements meeting.

Version R0.3 24/08/2011 Andrew Cannon. Revised following group and client requirements meetings.

Version R0.4 26/08/2011 Andrew Cannon. Minor revisions and an additional mockup added. Deliverable A submission version.

Preface:

This document addresses the requirements of the Sheep Methane Production Calculator project. The intended audiences for this document are the designers and the client of the project. This project is being completed by CITS3200 Group B for Dr. Samantha Bickell of the School of Animal Biology at UWA.

Target Audience:

Client, Developers

Group B Members:

Andrew Cannon (20175501)
Bowie Chung (10996822)
Darren Huang (20148398)
Hao Huang (10912626)
Ashwina Hurdowar (10913824)
James Nelson (20161432)

Meetings:

08/08/2011 – 12:51pm – Lab 2.03 – Group Meeting 1
10/08/2011 – 2:00pm – Reid Library Study Room – Group Meeting 2 (with client)
15/08/2011 – 12:52pm – Reid Library Study Room – Group Meeting 3
18/08/2011 – 3:25pm – Reid Library Collaboration Area – GUI Team Meeting 1
22/08/2011 – 1:00pm – Reid Library Collaboration Area – Requirements Meeting 1
24/08/2011 – 12:00pm – Reid Library Collaboration Area – Requirements Meeting 2
24/08/2011 – 2:00pm – CSIRO Floreat – Client Requirements Meeting

Milestones:

- 26/08/2011 - Initial RAD (Deliverable A) due. (Project Management)
- 16/09/2011 - Final RAD (Deliverable B) due. (Project Management & Developers)
- 21/10/2011 - Client handover (Deliverable C) of completed system with associated documentation. (Developers & Testers)
- Week of 31/10/2011 – Final presentation. (All)

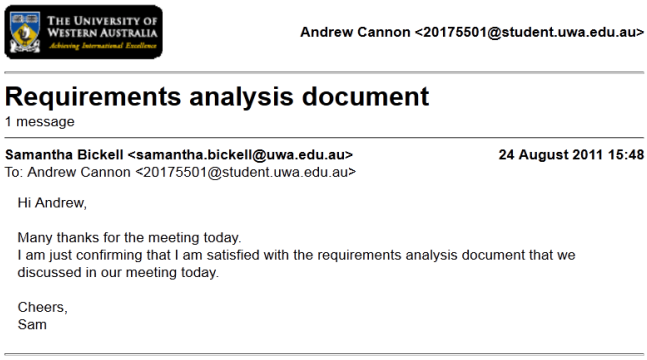
Sign Off:

I have reviewed and agree with the contents of this requirements document as a basis for the project to be completed.

Client Signature **Date**

---------- ----------

This requirements document is not signed by the client, as it has been revised since being presented to the client (based on the discussions of that meeting). However, the client has indicated her approval with the direction of this document via email.



THE UNIVERSITY OF WESTERN AUSTRALIA
Achieving International Excellence

Andrew Cannon <20175501@student.uwa.edu.au>

Requirements analysis document

1 message

Samantha Bickell <samantha.bickell@uwa.edu.au> **24 August 2011 15:48**
To: Andrew Cannon <20175501@student.uwa.edu.au>

Hi Andrew,

Many thanks for the meeting today.
I am just confirming that I am satisfied with the requirements analysis document that we discussed in our meeting today.

Cheers,
Sam

1.0 General Goals

The project is being carried out for Dr. Samantha Bickell (the client) by CITS3200 Group B (the developers). Dr. Bickell is working on a project for CSIRO and the School of Animal Biology at UWA breeding sheep that produce low amounts of methane in order to reduce the greenhouse gas emissions produced by agriculture. A vital part of this endeavour is measuring the amounts of methane emitted by particular sheep within a 23 hours period. This is done by performing extensive calculations on data obtained from various instruments. These calculations are currently performed largely manually using an Excel spreadsheet. The goal of this project is to develop an application that will automatically perform the necessary calculations upon being provided with the appropriate raw data files. It is intended that this will not only make the calculation process substantially faster, but also more reliable, by eliminating the possibility of human error in extensive manual calculations. The system must run on the existing lab computers and be usable for researchers without computer science backgrounds.

2.0 Current System

The experimental setup consists of 4 airtight respiration chambers, consisting of a pen for an individual sheep placed within a plastic container. A sheep is placed within each chamber for approximately 23 hours when an experiment is running. The air passing through each chamber is periodically measured for methane content by a gas chromatography (GC) system connected to a lab computer. The GC system produces numerous data files on this computer when it is running.

Numerous sensors around and inside the respiration chambers collect temperature and pressure information, which is logged and manually downloaded to a second lab computer on a weekly basis providing temperature and pressure data files.

Information on the volume of food eaten by each sheep is sent by Bluetooth to a third lab computer when the system is running to produce food consumption data files.

The various data files are transferred via shared network drives or USB drives to a single computer (usually the computer running the GC system). The GC data files are processed using an Excel macro and imported into Excel. The remaining data files are directly imported into Excel. Numerous calculations are performed on the collected data in Excel.

The calculations ultimately determine the volume of methane (in litres) produced by each sheep within the 23 hour period. Additional calculations may be performed to determine the methane produced by each sheep on an hourly basis and the quantity of food eaten on an hourly basis. This data is made available on a shared network drive. Occasionally, raw data from the GC system may be examined using a software program called GC Solutions on the GC computer to gain insight into the underlying data if the final result seems unusual.

3.0 Proposed System

3.1 Overview

It is proposed that an application be developed to run on a lab computer or other system that is capable of accessing a complete set of raw data files. It will allow the user to specify a set of data files and then perform all the required calculations, removing the need to import data into Excel and manually perform the computations. The results of the calculations will be displayed to the user and written to a user specified file. The application will be written in Java.

3.2 Functional Requirements

3.2.1 List of requirements

The system will be used by members of the experiment research group. The following functions are required. The priority of each functional requirement from 1 (essential) to 5 (outside the scope of the project) is given.

3.2.1.1 Calculate total methane emitted in 23 hour period (Priority 1)

When provided with a set of data files, the system will calculate the methane produced by each sheep within the 23 hour period, display this information to the user and write it to a specified comma separated value (CSV) file.

3.2.1.2 Specify total experimental time (Priority 1)

The system will allow the user to specify how long each sheep was within a particular respiration chamber. This information will be used in the methane calculations.

3.2.1.3 Calculate methane emitted on an hourly basis (Priority 2)

When provided with a set of data files, the system will calculate the methane produced by each sheep for each hour it was within its respiration chamber, display this information to the user and write it to a file.

3.2.1.4 Determine food consumed on an hourly basis (Priority 2)

If a set of food data files is present, the system will determine the quantity of food consumed by each sheep for each hour it was within its respiration chamber, display this information to the user and write it to a file. The system will note the absence of food data files if they are not present, but still proceed with the remaining calculations (as food consumption information is not used in methane production calculations – it is collected for reference and comparison purposes).

3.2.1.5 View raw food consumption data (Priority 3)

The system will display the raw food consumption data, highlight the representative value for a particular hour and its neighbours and allow the user to select an alternative value.

3.2.1.6 Display food-methane table (Priority 4)

The system will be able to display a table comparing the food consumed against the methane produced on an hourly basis and print this information to a file.

3.2.1.7 Additional respiration chamber support (Priority 1)

The system will be able to support an experimental setup involving arbitrarily many respiration chambers, assuming the data files for the additional chambers follow the same format as the first 4 chambers. The client specified that it is vital that the system support an 8-chamber setup (as they will likely switch to such a setup soon) in addition to a 4-chamber setup, but support for higher numbers of chambers is less important.

3.2.1.8 Display food-methane graph (Priority 4)

The system will be able to display a graph comparing the food consumed against the methane produced on an hourly basis and output it to a file.

3.2.1.9 Record historical data (Priority 2)

The system will save the results of computations to an internal database allowing previous results to be viewed.

3.2.1.10 Display historical methane production table (Priority 4)

Using information from its internal database of historical calculations, the system will be able to display a table showing total methane production (within 23 hour periods) over time for numerous sheep.

3.2.1.11 Display historical methane production graph (Priority 4)

Using information from its internal database of historical calculations, the system will be able to display a graph showing total methane production (within 23 hour periods) over time for numerous sheep.

3.2.1.12 Specify operations to be performed on data sets (Priority 4)

The system will allow the user to specify arithmetic operations to be performed on particular data sets.

3.2.1.13 Compare data sets graphically (Priority 4)

The system will allow users to plot one data set against another on a graph and output the graph to a file.

3.2.1.14 Semi-automatic importation of new data files (Priority 1)

The system will remember the folders where data files were previously imported from and automatically select new files in these folders for importation and processing. The user should be prompted to confirm this selection before the system processes the new data files.

3.2.1.15 Annotate data sets (Priority 3)

The system will allow the user to specify, view and edit annotations to accompany imported data sets.

3.2.1.16 Display computation results on a web site (Priority 5)

The system will display results of computations performed on a web site. This functionality will not be implemented and is beyond the scope of the project, as CSIRO already has the infrastructure to distribute data to researchers over its network using shared network drives.

3.2.2 Table of client values

This section indicates the value that the client assigns to particular functional requirements, by listing the value (in dollars), out of a total of 100 dollars, that she would assign to particular requirements. See the previous section for requirement descriptions.

Number	Name	Value
3.2.1.1	Calculate total methane emitted in 23 hour period	$15
3.2.1.2	Specify total experimental time	$15
3.2.1.3	Calculate methane emitted on an hourly basis	$8
3.2.1.4	Determine food consumed on an hourly basis	$8
3.2.1.5	View raw food consumption data	$5
3.2.1.6	Display food-methane table	$1
3.2.1.7	Additional respiration chamber support	$15
3.2.1.8	Display food-methane graph	$1
3.2.1.9	Record historical data	$8
3.2.1.10	Display historical methane production table	$1
3.2.1.11	Display historical methane production graph	$1
3.2.1.12	Specify operations to be performed on data sets	$1
3.2.1.13	Compare data sets graphically	$1
3.2.1.14	Semi-automatic importation of new data files	$15
3.2.1.15	Annotate data sets	$5
3.2.1.16	Display computation results on a web site	$0

3.2.3 Requirements Ranking

This table lists the client value of each requirement (from the previous table), the estimated time to completion (ETC) of each requirement in person hours, the value estimate ratio (VER) of each requirement in dollars per person hour (computed by dividing the client value by the ETC) and a ranking specifying which requirements add the most value for the least labour (determined by VER in descending order).

Name	Value	ETC (person hours)	VER ($/person hour)	Rank
Calculate total methane emitted in 23 hour period	$15	40	0.38	8

Specify total experimental time	$15	12	1.25	2
Calculate methane emitted on an hourly basis	$8	10	0.8	4
Determine food consumed on an hourly basis	$8	6	1.33	1
View raw food consumption data	$5	30	0.17	=11
Display food-methane table	$1	6	0.17	=11
Additional respiration chamber support	$15	16	0.94	3
Display food-methane graph	$1	15	0.07	13
Record historical data	$8	20	0.4	7
Display historical methane production table	$1	6	0.17	=11
Display historical methane production graph	$1	12	0.08	12
Specify operations to be performed on data sets	$1	30	0.03	=15
Compare data sets graphically	$1	30	0.03	=15
Semi-automatic importation of new data files	$15	30	0.5	=6
Annotate data sets	$5	10	0.5	=6
Display computation results on a web site	$0	NA	0	16

3.3 Nonfunctional Requirements

This section lists the non-functional requirements of the system, which specify constrains on how the system may satisfy its functional requirements.

3.3.1 User Interface and Human Factors

The users of the system will be the members of the research group. Given the highly specialised nature of the experiment, it is expected that users will require specialised training to master the experimental setup including the hardware and software (including the system being developed) being used. However, these users will not necessarily have computer science backgrounds.

As a guide, it is expected that users will be able to use the core functionality of the system (specifying data files and writing computation results to a file) and be comfortable accessing remaining features (possibly with reference to a user manual) after one hour of training. This requirement is likely to be difficult to verify given the timeline of the project, but usability issues will be addressed by showing the client and users iterative prototypes of the interface as it is being developed to ensure they are comfortable using the final version.

3.3.2 Documentation

At the completion of the project, the following documentation describing the system will be provided by the developers to the client. The documentation specified below meets or exceeds the requirements for Deliverable C.

- Complete Java source code documented using Javadoc.
- Automated tests (written in Java) with accompanying documentation.
- Instructions on building the application.
- Instructions on installing the application for users with non-technical backgrounds.
- User manual describing how to use the functionality offered by the application intended for potential users.
- Description of the system architecture and information on how to implement anticipated functionality described in Section 3.3.8, intended for future developers seeking to extend the system.

3.3.3 Hardware Consideration

The application will need to run on the existing lab computers. These computers run Windows XP. The GC computer has a 3.2Ghz Pentium 4 processor and 1GB RAM. It is not anticipated that the application will strain the target computers in terms of processor usage, memory or disk storage.

3.3.4 Performance Characteristics

System performance is not expected to be a major issue, given the scope of the calculations to be performed. As a guide, it is expected that the startup time for the application (when the host computer is loaded and no other programs are running) will be 5 seconds or less and that it will take 30 seconds or less to perform the required computations for the 4 respiration chambers. These estimates are provisional and may be revised upwards. There is substantial latitude for upwards revision of these estimates given the relative infrequency that calculations are performed and the fact that any reasonable automated system would be substantially faster than the current manual system.

3.3.5 Error Handling and Extreme Conditions

The system should be capable of operating without a set of food consumption data files (as this information is compared with the methane calculations, not used within them).

The system should notify the user if data files are invalid or corrupt.

A set of data files normally contains information on four sheep in four respiration chambers. If data is corrupt or missing for one or more chambers, the system will still be capable of performing calculations on the data from the remaining chambers.

The user should be notified if they attempt to have calculation results written to a pre-existing file.

3.3.6 System Interfacing

The system processes data files obtained from measurement systems previously described. These data files are assumed to have a fixed format. The system will need to be modified if any data file formats change.

The system outputs the results of computations to user specified files. The system documentation will describe the format of outputted files.

3.3.7 Quality Issues

The application will be opened by users when they wish to perform computations and then closed. It will not be running continuously, so discussions of acceptable downtime are not applicable.

The application must be capable of running on present and future lab computers. It cannot be guaranteed that the application will be capable of running on unspecified future hardware, but implementing it in Java maximises the likeliness that it will be able to run on both present and future lab computers.

3.3.8 System Modifications

The system will be designed with the following assumptions.

- Raw data file formats will change.
- Additional data files may need to be read.
- Additional types of calculations will need to be performed.
- Existing calculations will need to be modified, for example, to change the units of measurement used in a particular calculation.
- Calculation results will need to be output to files in different formats.

The design of the system should therefore anticipate expansion in these areas.

3.3.9 Physical Environment

Not applicable.

3.3.10 Security Issues

Not applicable.

3.3.11 Resource Issues

Not applicable. It is assumed that the files output by the system will be backed up according to CSIRO's existing backup procedures.

3.4 Constraints

The system must run on the existing lab computers. Given the scope of the computations to be performed, this is not expected to be a problem.

The application must be capable of operating without network access, as it was specified by the client that lab computers may not have network access in the future.

The client did not specify constraints on the programming language, development environment or the use of libraries. The developers have decided to implement the application in Java due to the team's familiarity with it and its cross platform nature. The Netbeans development environment will be used due to its features for implementing Java Swing graphical user interfaces.

3.5 System Model

3.5.1 Scenarios

3.5.1.1 Running the application for the first time

Alice places the application CD in a lab computer and copies the application folder to its hard disk. She double clicks on the application file which starts the application and goes to import data. She selects GC, pressure, temperature and food data files by navigating to these files on the computer's hard disk and on shared network drives. She clicks on the button to run calculations, specifies the time that each sheep was in a particular respiration chamber and then a short time later, the methane produced by each of the 4 sheep in the 23 hour period is displayed. The data looks fine, so she specifies a file where this information (in CSV format) should be written by navigating to a location on a shared network drive and then closes the application. She then opens Excel and imports the CSV data for analysis.

3.5.1.2 Running the application after the first time

Alice double clicks on the application file to start it and goes to import data. The application has remembered the location of the data files she accessed the last time she ran the program and as the new data files are in the same location, she can quickly select the files she wants to process. She clicks the button to run the calculations, specifies how long each sheep was in each respiration chamber and then the methane produced by each of the 4 sheep is displayed. She then displays a table of the methane produced by one of the current batch of sheep on an hourly basis. Satisfied with her examination of the data, she specifies that it should be output to a CSV file. This can be done quickly, as the system remembers the location of the last output report that she saved. She then closes the application.

3.5.1.3 Running the application without network access

Alice downloads the data files from the temperature and pressure loggers onto the old laptop, which still has a serial port with which to access this hardware. She copies the data produced to a USB key, then shuts down this laptop. She then access the computer for recording sheep feed volumes and copies the feed data file to her USB key. Finally, she accesses the GC computer. She copies the data files from the other two computers onto this computer's hard drive and runs the application. She then runs various calculations on the data files as in the previous two scenarios. Finally, she has the application copy its calculation report (in CSV format) onto her USB key. Finished, she shuts down the GC computer and takes the USB key to her office to analyse the data.

3.5.2 Use Case Models

3.5.2.1 Actors
3.5.2.2 Use Cases

3.5.3 Object Models

3.5.3.1 Data Dictionary
3.5.3.2 Class Diagrams

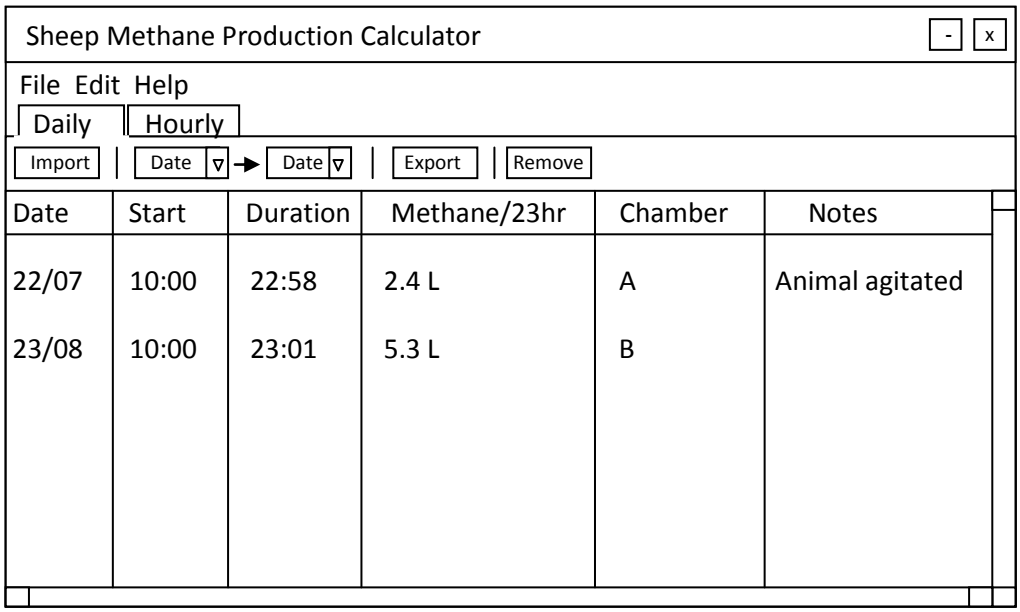
3.5.4 Dynamic Models

3.5.5 User Interface

3.5.5.1 Navigational Paths

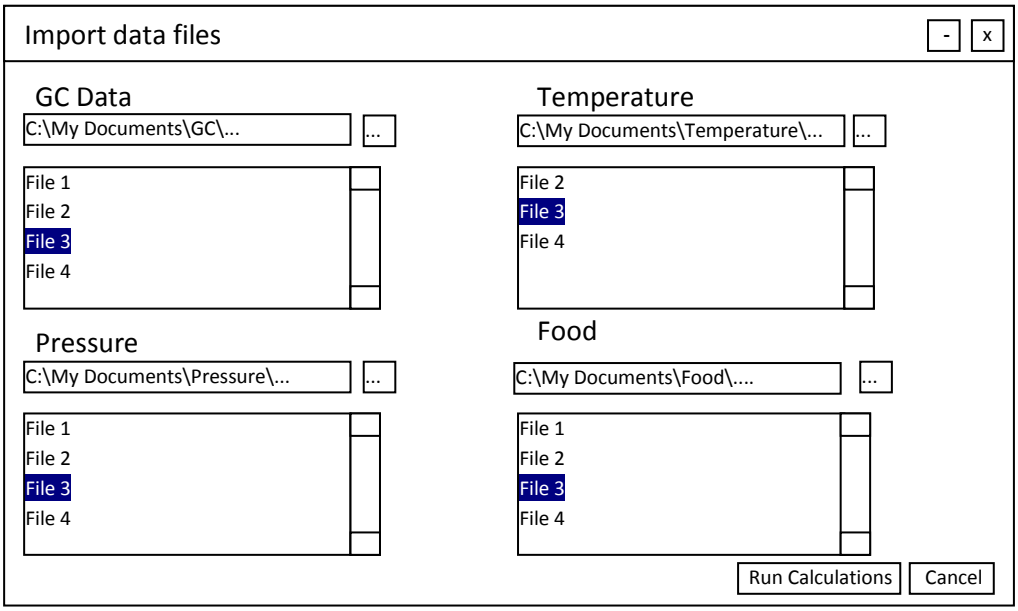
3.5.5.2 Screen Mockups

3.5.5.2.1 Main Window



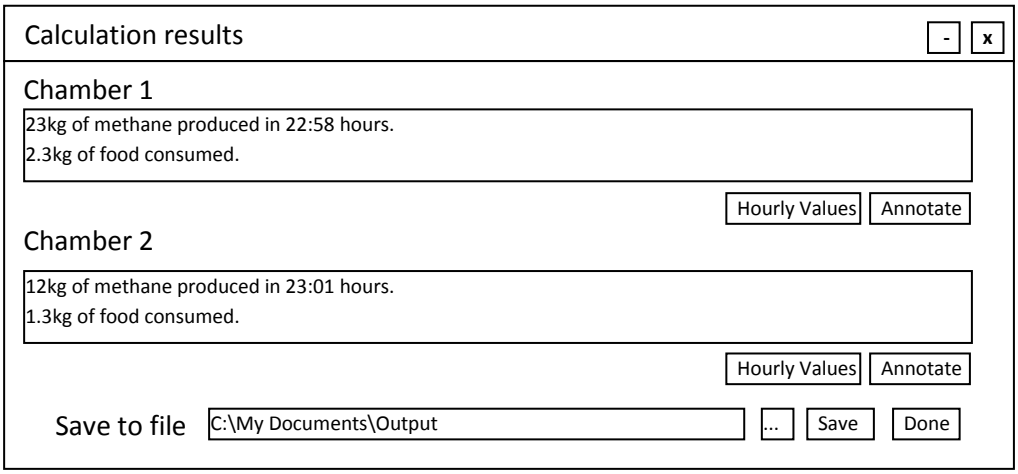
This shows the main window of the application. A table of data from historic experiments is shown. The user is able to specify the date range of experiments to examine. There are tabs to examine daily and hourly methane production. There are buttons to import new data or remove rows from the table. Finally, there is a button allowing the data to be exported to a CSV file.

3.5.5.2.2 Data Files Importation Screen



This dialogue is used to import data files of different types (semi-automatically) for processing and calculations. The first time that the user accesses the system, the locations of data files must be manually specified. On subsequent occasions, the system remembers the folders where data files were previously imported from, and displays a list of files in those folders. It automatically selects the most recently added files. The user is then able to glance at these to verify that they are correct and then select "Run Calculations".

3.5.5.2.3 Calculation Results Screen



This dialog displays the results of computations for different respiration chambers after they have run. The user can additionally view the methane produced on an hourly basis, add an annotation to a particular result and specify a CSV file into which results should be written.

4.0 Glossary

Application. Proposed Java application to automate the calculations required to determine the amount of methane produced by particular sheep within a 23 hour period from a set of raw data files.

Client. Dr. Samatha Bickell of the School of Animal Biology at UWA.

CSV. Comma separated values (file).

Data files. The raw data files produced by the GC system and other instruments used to calculate the amount of methane produced by particular sheep.

Designers. See **developers**.

Developers. CITS3200 Group B in Semester 2, 2011.

Gas chromatography (GC). Process to determine the methane content within a particular respiration chamber at a particular point in time. The GC system produces numerous data files on a particular lab computer.

Lab computers. The computers currently used by Dr. Bickell and her colleagues in their laboratory to perform calculations related to sheep methane production.

Research group. Colleagues of Dr. Bickell at CSIRO and UWA working on the sheep methane experiment.

Respiration chamber. Pen for an individual sheep placed within an airtight container used in the client's experimental setup. The methane in each chamber is periodically measured by the GC system.

System. See **application**.

UWA. The University of Western Australia.