FOALARM

FOALING ALARM

FUNCTIONAL SPECIFICATION

DAVID KELLY | C00193216

Bachelor of Science (Honours) Software Development

Supervised by Nigel Whyte. Date: 14th November 2017.

TABLE OF CONTENTS

ABSTRACT

The purpose of this document is to outline the functional specification of my 4th year Software Development project. The following sections will detail business requirements, proposed system architecture and core functionality; additionally, a use case diagram, brief user stories and non-functional requirements. A development plan is provided also.

INTRODUCTION

Objective

This project aims to develop a foaling prediction system composing of a: wearable foaling alarm; central hub; cloud platform for data processing and storage; and, a web application for realtime alarm notifications and crowd sourcing mare monitoring.

User Requirements

Attended foal deliveries are essential to ensuring the safety of both mare and foal. Traditionally, breeders anticipate sleepless nights in the days or weeks prior to delivery as 80% of mares will foal between 11 pm and 3 am. Dytocias (difficult births) account for 10% to 70% of all foal births, depending on breed, where rapid appropriate intervention is critical for both mare and foal survival. Despite this significant statistic the majority of dytocias are easily alleviated by foaling managers, signifying the importance of attended foal deliveries.

As outlined in this project's research documentation, the use of sensor technology for foaling prediction has been available since 1987. These products utilize a mare's tendency to lie in lateral recumbency moments prior to and during foaling. While multiple variations of this concept are available to breeders; visual monitoring remains the most common method of ensuring attended deliveries.

Despite the plethora of foaling prediction technology and practices none attempt to converge traditional mare visual monitoring and sensor technology in a single cloud platform for mare owners and foaling managers.

Project Definition

FOALARM is an equine foaling prediction system. The system is comprised of:

- Wearable Foaling Alarm
- Central Hub
- Cloud Backend
- Web Application

The wearable foaling alarm will be attached to an expectant mare prior to her anticipated due date. The device will be secured in a plastic housing and attached to the chin strap of a halter using a water resistant material pouch. The wearable will measure locomotor activity using a three axis accelerometer to determine when the mare is positioned in lateral recumbency. Upon identification of a defined period of lateral recumbency the wearable will transmit the data to the central hub via RF transceiver.

The central hub will be connected to the internet and be located in close proximity to expectant mares (i.e. the wearable foaling alarms). The hub will receive data from broadcasting wearable foaling alarms. Upon receipt of data from a wearable, the hub will upload the transmission to a cloud database.

Cloud-based server processes will observe database changes by connected hubs as foaling events. Identified foaling events will generate realtime user alerts via SMS, email and web app push notification.

Mare owners will use the web application to: register FOALARM hardware components; assign wearable devices to expectant mares; set point of contact for SMS and email foaling alarm notifications; enter pre-existing IP camera links for embedded visual mare monitoring inside the web application; view all active wearable devices and IP camera links for crowd sourcing mare monitoring; and receive in-app realtime alarm notifications.

Additional Functionality

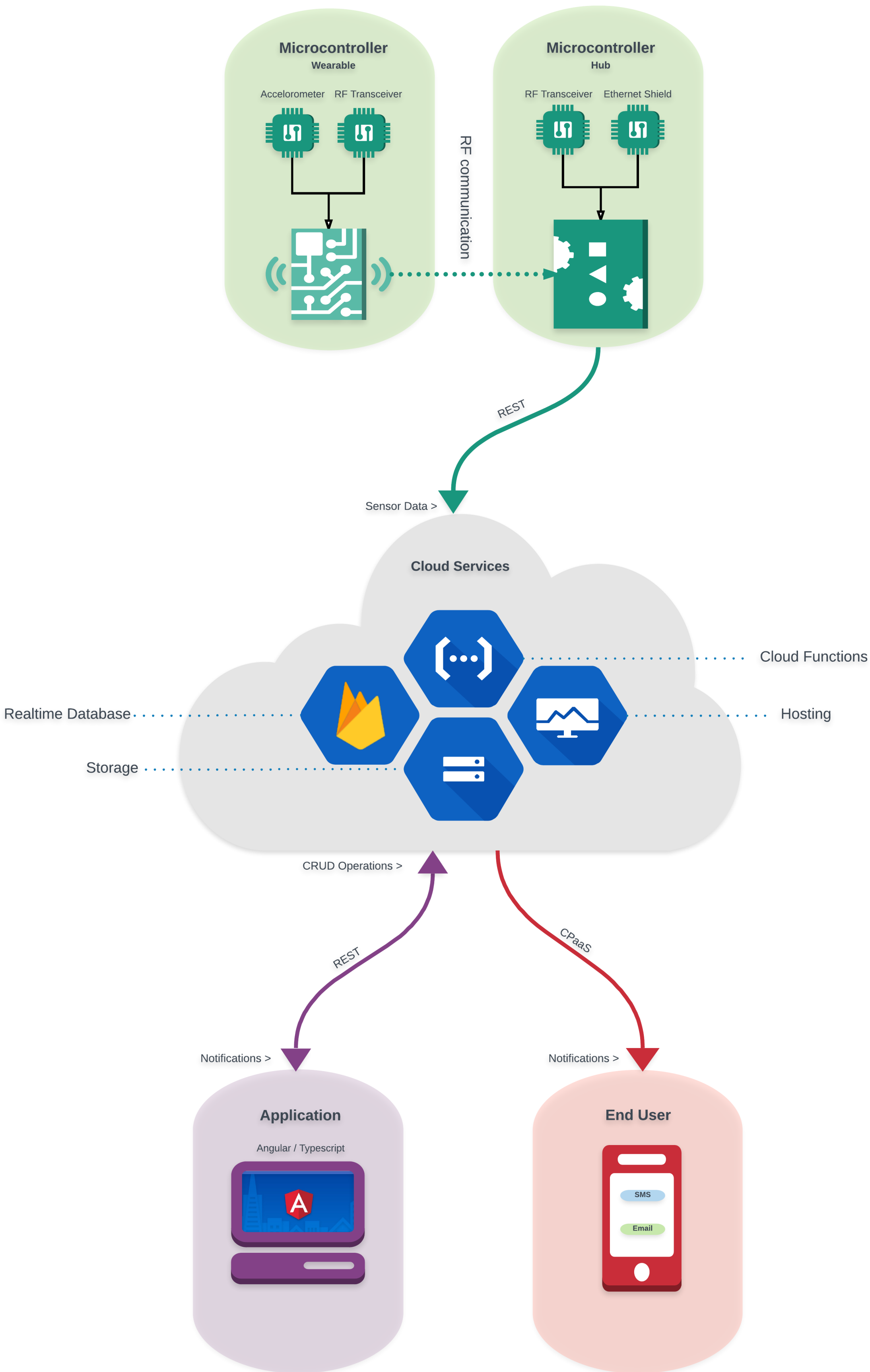
The user will toggle a switch on the wearable device to enable sensor data streaming. The wearable foaling alarm will be attached to an expectant mare approximately **two weeks** prior to her anticipated due date. The wearable will measure locomotor activity using a three axis accelerometer and continuously stream the data to the central hub.

The central hub will function as usual.

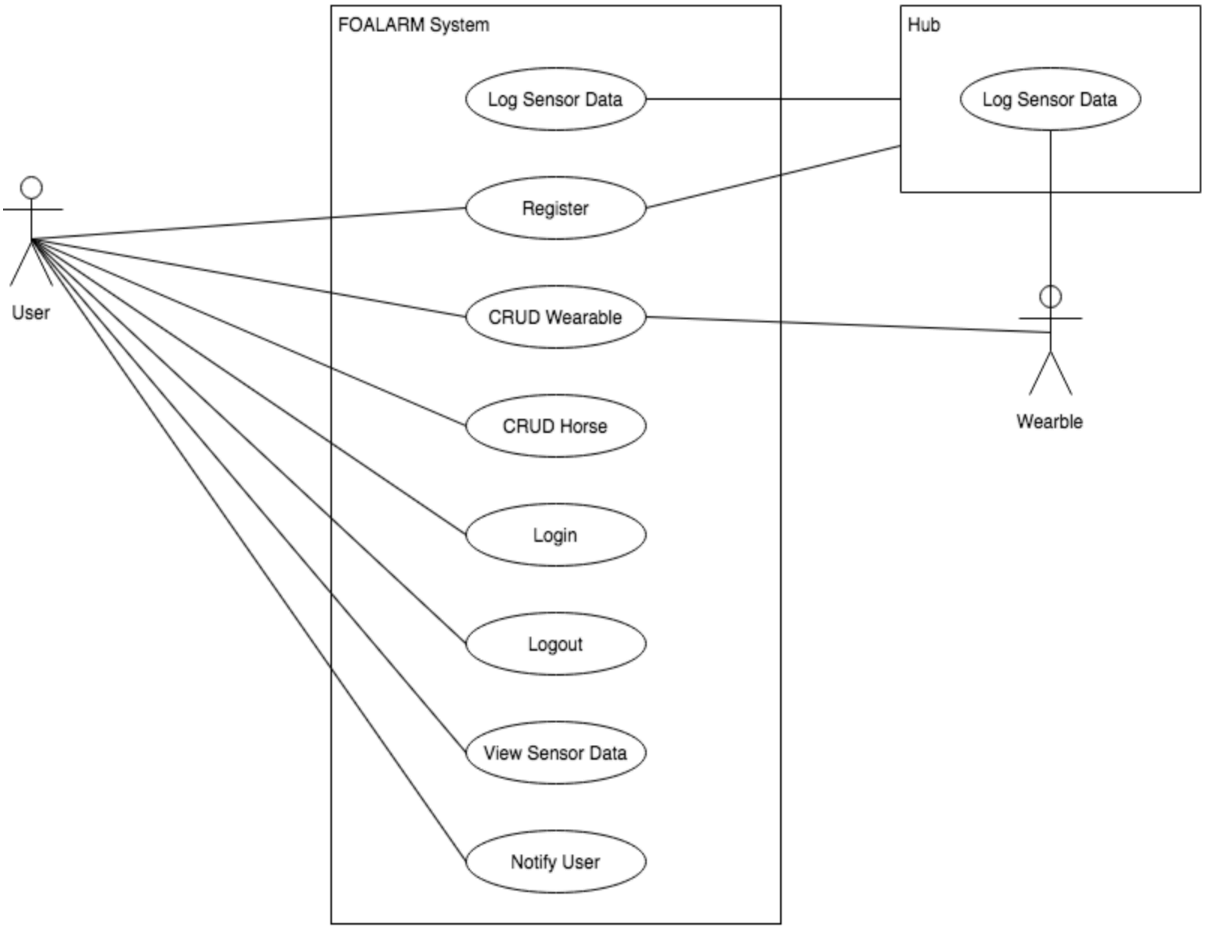
Locomotor activity data will be processed on the cloud-based server to identify foaling events. Identified foaling events will generate realtime user alerts via SMS, email and web app push notifications.

The application will function as usual.

SYSTEM ARCHITECTURE



USE CASE MODEL



BRIEF USE CASES

Log Sensor Data

Actors

Wearable, Hub, System

Description

This use case begins when a Wearable is about to log sensor data. The Wearable transmits the data to the Central Hub. The Central Hub, connected to the database, receives the data. The Central Hub enters the data to the database.

Notify User

Actors

User, Wearable, System

Description

This use case begins when the System receives new sensor data. The System sends an SMS and email notification to the respective wearable's point of contact. The System updates the Wearable status and saves it to the database. The status change is pushed to the web application notifying connected clients.

CRUD Wearable

Actors

User, Wearable, System

Description

This use case begins when an authenticated User is registering a Wearable foal alarm. The User selects the add alarm functionality on the web app. The User enters the Wearable serial id, point of contact, IP camera link, assigns the wearable to an existing horse and presses submit. The system validates the input information and saves it to the database. The User is redirected to the alarms view where the newly created Wearable is displayed.

CRUD Horse

Actors

User, System

Description

This use case begins when an authenticated User is registering a horse. The User selects the add horse functionality on the web app. The User enters the horse's information and presses submit. The system validates the input information and saves it to the database. The User is redirected to the horse view where the newly created horse is displayed.

View Sensor Data

Actors

User, Wearable, System

Description

This use case begins when tan authenticated User wishes to view a Wearable’s sensor data. The User selects the Wearable from the alarms view on the web app. The System retrieves the selected Wearable’s sensor data from the database and suitably displays it on screen.

Register

Actors

User, Hub, System

Description

This use case begins when the User select the register functionality on the web app. The User enters their name, email address, password, Hub serial id and selects submit. The System validates the input information and saves it to the database. The User is redirected to the authenticated application view.

Login

Actors

User, System

Description

This use case begins when a registered User requires access to the System. The user selects the login functionality on the web app. The user enters their email address, password and presses submit. The System validates the input information. The User is redirected to the authenticated application view.

Logout

Actors

User, System

Description

This use case begins when an authenticated User intends to log out of the System. The User selects the logout functionality on the web app. The System logs the User out of the system and redirects the User to the unauthenticated application view.

SUPPLEMENTARY SPECIFICATION

The purpose of this section is do detail the supplementary requirements that are not captured in the user stories of the FOALARM use case model.

Functionality

- All system functionality will be remotely accessible.
- The wearable and hub will ideally be located in proximity or within 500 feet unobstructed line of site.
- The central hub will require a continuous internet connection for data transfers.
- Connected users will be authenticated before being permitted to each application view.

Usability

- A first time user should be able to register and configure a FOALARM system in 10 minutes.
- A registered user should be able to login to the system in 5 seconds.
- The application interface will be fully responsive and will correctly render to desktop, tablet and smartphone device viewpoints.
- The application's interface should remain consistent throughout all views.
- Application functionality will be browser agnostic.
- Appropriate help, instructions or hints will be provided for all system functionality.

Reliability

- All database operations will be atomic and indivisible.
- If a server connection is not available appropriate feedback will be provided to the user.
- The server will be able to handle 1000 requests per second without negatively affecting performance and usability.
- The system should correctly store and retrieve all information provided by the user.

Performance

- The web client should take no more than 5 seconds to complete any client-server transaction.
- In the event that any client-server transaction takes longer than 5 seconds the web client will display an informational message to the user notifying him/her of the transaction's progress.
- In the event that the web client cannot successfully establish a connection to the server an informational error message will be displayed to the user.

Supportability

- The application should be supported by multiple browsers with functionality, performance and usability remaining consistent.

Security

- Only registered and authenticated users will be permitted access to the web application's functionality.

DEVELOPMENT STRATEGY

The FOALARM project's development strategy is outlined in the following iterative aspirations.

Iteration 1

The first iteration will focus on the FOALARM system's core functionality, which is logging sensor data from a wearable to the cloud-based database and generating realtime notifications, as captured in the Log Sensor Data and Notify User use cases respectively.

Log Sensor Data: As outlined in the brief use case, this functionality will involve the Wearable, Hub and System (specifically the cloud-based database). The Wearable, consisting of an accelerometer and radio module will continuously read the sensor data to identify a predefined period of time when the mare is lying in lateral recumbency. This event will invoke a data transmission via RF to the Hub. The central Hub will receive the data transmission, identify the broadcasting wearable and generate a new data entry for the database. The Hub will be connected to the cloud server (WiFi and Ethernet capabilities will be available) to push the new data entry to the database.

Notify User: This will consist or realtime notifications to the Wearable owner's specified points of contact. New data entries to the database by connected FOALARM hubs will invoke a realtime notification service. The service will subscribe to the respective database nodes where changes to that node will generate SMS and email notifications alerting the end user that a particular mare has been identified to be in labour.

The design and build of hardware components will be a significant aspect of this use case. The Wearable will be built using Tinycircuits microcontroller technology to reduce the need for soldering and wiring. The Hub will consist of an Arduino microcontroller with built in WIFI and Ethernet capability. The Hub's RF transceiver will require minimal soldering and wiring.

To facilitate the aforementioned use cases database modelling is required to establish the JSON object structures for storing and retrieving wearable, hub and user attributes. Due to the nature of Firebase Realtime Database queries it is recommended to model the database according to application views. Therefore, basic app functionality, as captured in the Register, CRUD Alarm and CRUD Horse use cases will be required to determine a suitable data model.

Register: This will consist of a registration form that a new FOALARM system user will complete to configure their FOALARM components and gain access to the application's functionality. The registration form will request standard user data as well as a FOALARM Hub serial id. This will be used to identify the Hub in the database and authenticate future Hub communication with the cloud server. Upon completion the user will be permitted access to the FOALARM system.

CRUD Alarm: This will consist of a form to configure a FOALARM Wearable. The user will be required to enter a FOALARM Wearable serial id, point of contact and IP camera link (if available). Finally, the alarm must be associated with an existing horse.

CRUD Horse: This will consist of a form to enter expectant mares into the system. The user will specify a number of attributes (such as, but not limited to: name, expected due date, image). The horse can now be associated with a Wearable prior to her expected due date.

Documentation, testing, integration and regular deployment will be continuous throughout all iterations.

Iteration 2

The second iteration will involve the development of the applications functionality as captured in the Login, Logout and View Sensor Data user stories.

Login: This will consist of a form to authenticate a registered user to the FOALARM web application. The user will enter their email address and password to login. Successfully authenticated users will be permitted access to the FOALARM system.

Logout: This will consist of a button that the user will select to end the current session logout of the FOALARM web application. Upon successful logout the user will be redirected to the Login screen.

View Sensor Data: This will be some graphical representation of individual wearable data transmissions.

The CRUD Alarm and CRUD Horse use cases will be further developed and extended to enable sharing connected wearable devices with all authenticated users. The Notify User use case will be further developed to include push notifications. This will provide the application with the functionality of crowd-sourcing mare monitoring with realtime in-app notifications.

Locomotor activity analysis will be limited to lateral recumbency identification in the first iteration. This method remains to be the most widely used and trusted; and the earliest predictor on immanent foaling as outlined in this project's research. The second iteration will explore the implementation of foaling prediction using continuous streaming of accelerometer data. This will involve further development of the Wearables functionality: the ability to manually switch between lateral recumbency identification and data streaming. The Hub will function as before, pushing data transmissions to the database. Server side data analysis will be implemented to identify immanent foaling events based on collected locomotor activity. Identified foaling events will be saved to the database allowing the notification service and web application to function as before.

Iteration 3

In the third iteration, if time permits, it is planned that the following functionalities be achieved:

- Develop algorithms for equine locomotor activity to identify and classify movement patterns in predicting immanent foaling.
- Develop remaining application features.

CONCLUSION

The proposed architecture and core functionality of the FOALARM system, as is currently know, is outlined in this specification. The brief use cases capture primary working features that will distinguish this project from existing products. A development plan will aid in the time management and in ensuring a functioning and testable prototype is incrementally delivered throughout the course of development.