SOFTWARE ENGINEERING PROGRAMME
UNIVERSITY OF OXFORD
www.softeng.ox.ac.uk

ASSESSMENT

Student: Nicholas Drake

Course: Embedded Software and Systems

Date: 22nd January 2024

Grade: 76

Andrew Markham April 15, 2024

REPORT

Overall: This is a very strong report that clearly demonstrates mastery of the fundamentals of embedded software and systems. It details the step-by-step design of a pet-tracker/display. What was particularly strong was deep discussions about component and method selection.

Requirements & Specifications: The requirements were specified adequately - it would have helped to have some deeper formalism (e.g. ECA) especially as you have provided a state-transition table on p11.

Architecture and design: This was a very well argued and motivated section, particularly the discussion about the microcontroller selection. What was good here is that you explicitly linked up the requirements to the justification of the choice of the nRF9 series, and how you narrowed it down from generic platforms down to a vendor specific solution. I also liked that you dug into the datasheet to demonstrate how to set the prescaler to obtain the correct sampling rate. Similarly, your compare-contrast to motivate the use of an OLED display is excellent.

Implementation: This was done well. I liked that you thought more broadly about aspects such as security i.e. that the location of a pet can also indicate the location of the owner, and used this to justify the use of an RTOS such as Zephyr. The implementation was clear through the use of the state-table (ECA) pattern and the hierarchical state machine.

Testing: This was a good overview of how testing could be achieved - it perhaps would have helped to show a small mock example especially wrt to your use of zephyr allowing POSIX style mocks.

Deployment: Here, you discussed post-release issues (logging, OTA updates) competently, as well as thinking clearly about risks and their mitigation.

Optimization: This section was done well with a solid discussion on how to optimize power consumption. It could have been extended by thinking about pain-points for the product e.g. the need to recharge the collar - could you reduce this by harvesting energy e.g. through pet motion or solar power? Could you reduce the cost of fabrication of your device through optimization? Would it help (power or functionality wise) to also have a BLE device - you

could then obtain a GPS fix by proxy through a connected smartphone, and avoid having to use LTE to send data? What would be the cost/complexity implications?