D4 Form 5: Project Completion Form

Cost Estimates

Fixed Costs:

Hardware and Software development costs.

Estimation of 670 Person hours on D4, including preliminary research, design meetings, planning and development of the system, integration, advertising, trade =£67,000.

Component Costs
All components acquired for D4 (whether used or not) = £286.19 Approx.
Taken from Purchase Approval Request form, and additional power components.
Also fixed server hosting costs, scalable to a large userbase. Estimation of £100 per year.
=£390.

Conformance testing
CE mark
=£2000.

Development and Manufacturing Overheads.
Fixed cost given to us.
=£100,000

Total = £169,390

Variable Costs:

Manufacturing Component Costs

Cost of required components to build the system.
Ideally, after prototype, develop PCB and not have loads of Dev boards / breakouts, then get that manufactured for much cheaper.
But in our case:
Components = £57.82
Assuming large quantities (discounts on some components).
Estimation for casing costs is around £1 per unit = £2 per system.
=£60 total.

Assembly Costs
Person hours it would take to assemble each unit on a production line. Includes collecting components, solder them together, flash ROMs, housing components in casing, package the complete unit. £20 per hour.
In real world, as stated before, get custom PCB will all the chips required already on it. Manufactured by a machine (soldered etc).

In our case, for each device, 2 mins to collect components, 15 minutes to solder components, 5 minutes to flash ROMs, 5 minutes to house in casing, 3 mins to package.
= 2 devices per hour per person.
20 people working on it = 40 systems (primary and secondary device pair) per hour.

£20 per hour = £400 per hour cost.

4 weeks production = 6400 systems ready to be shipped (5 day working week)
=£64,000 for 4 weeks of production (20 employees).

Another way of looking at it is £10 per system.

Calculating Profit:

System sold at £90 per pair (primary and secondary device).

90 (system cost) - 60(component cost) - 10(labour cost) = £20.

169,390 / 20 = 9119.5 units sold to start making profit.

Therefore 7166 units to be produced (99.5% yield).

Say if companies buy between 20-50 systems on average.

9166 / 35 = sold to 262 companies on average.

If price increase to £110, (increase by 22%), profit after 131 companies have purchased the product.

Design Changes

Analogue circuitry: the original plan was to do more analogue signal processing outside the MCU to rectify and sum the voltages from each sensor output. However, due to working on a single supply rail this would have required a significant amount of additional circuitry, and when the sensor was characterised, it was observed that the main changes in sensor readings were due to gravitational acceleration (meaning it was more effective to use the device as a gyroscope).

Pedometer: Initially the step detection was meant to be done using peak detection algorithm but due to simplicity Hysteresis algorithm with FIR filtering was used.

Heart rate monitor: The signal processing algorithm from library written by the manufacturer did not work very well and had to be replaced with our own.

LED indicator which indicates co-workers exercising moved from primary device to secondary device.

Server to website integration: Used node red to connect the server to the website. This used function nodes to parse the JSON data into local variables (stored in context), which could be later accessed and used.

COVID and D4 Remote Working

As planned, daily meetings were held, at first as a whole team, later within smaller groups. Team leader was updated about the progress and state of each module.
Projects have been assigned in a way to maximise the efficiency and utilise skills of each team member, allowing them to excel in their preferred area. We did well distributing the workload equally to keep everyone busy, while matching their preferences.
The budget was managed to ensure that everyone has a copy of the hardware that they needed to work with, such that all things that were produced by any team member would be reproducible by all other members in the team. Effective management of budget was implemented with input from the whole team. All programming updates are placed on the GitLab repository made for team PANDEMIC.
If any parts of the project became more difficult to implement than originally thought, this would be stated to the team leader as soon as possible such that team members could be reallocated to ensure the project is completed on time.
The integration team stated the requirements in terms of sensor interfacing code clearly, so that merging individual pieces of work wouldn't be an issue. Integration of products involved team calls with appointed members within the smaller modules and the integrating team, such that integration of entire design could be as quick and efficient as possible.

Actual Project Activities

Activity	Initials	Fri Am 26/2	Fri pm	Mon Am 1/3	Mon Pm	Tue 2/3	We d3/ 3	Thu 4/3	Fri Am 5/3	Fri Pm 5/3	Mon Am 8/3	Mon Pm 8/3
Test current draw of components under load	GP	X										
Analyse heart rate sensor datasheet and software library	PF	X										
Implement a program for getting time when activities were done	YDC	X	X									
Simulating analogue circuit.	H.S	X	X	X	X							
Analog circuit schematic	H.S	X	X	X	X							
Reading Raw values.	K.K		X									
Characterize accelerometer	H.S		X	X	X							
Integrate voltage regulator system with ESP32	GP		X									
Build test analog	H.S		X	X	X	X						
Get the ESP32 to successfully communicate with the sensor via I2C	PF			X								
Test current draw under most extreme draw conditions.	GP			X								
Analyze accelerometer data for Primary device.	K.K			X	X							

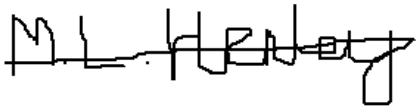
Test and debug if time is imported from the esp32 and exported to the server	YDC			X	X							
*Included timer libraries for attaching interrupts, setting up timer clock settings, and resetting the timer	HNG			X	X	X	X					
*Implement and debug a program by using timer libraries to detect inactivity for a fixed amount of time.	K.K					X	X	X				
*Implement Hall Effect Sensor.	BS		X									
Investigate possibility of improved charging system.	GP				X							
Produce LAN web server with MQTT AWS IoT	BS				X	X	X					
Develop a formula for calculating the “effort” for heart rate data	PF					X						
Add things to AWS IoT and connect the ESP32’s to the server	BS	X	X	X								
Get movement activity for primary device	K.K					X	X					
Get movement activity for secondary device	JW				X	X	X					
Add web client to AWS IoT and ensure MQTT protocol transmits data from ESP32 to web client and vice versa	BS					X	X	X				
Further improvements and debugging	PF						X					
Calculate points for primary device	K.K						X					
Calculate points for secondary device	JW						X					
Implement improved charging system if applicable.	GP						X					

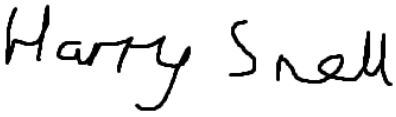
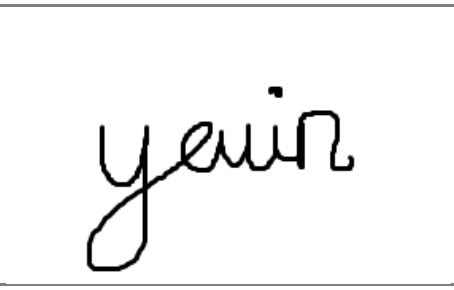
Integration of accelerometer with other modules in u controller	K.K							X	X			
*Integration of accelerometer on secondary device with other modules in uController	HS							X	X	X	X	
Debugging and improving GUI of website	MH								X			
Integrating entire system (sensors, server, website)	BS				X	X	X	X	X	X		
Building up a second pair of devices for demonstrating multiple users	HS								X	X	X	

Discrepancy in Project Activities

Kesavaram did code integration and debugging of the inactivity timer and notification functionality and Hock did setting up the timer functions and libraries. Initially, it was proposed that Harry Snell would do the digital processing for the hall effect sensor, however, due to timing considerations, Benjamin Sanati done the digital processing for the hall effect sensor while Harry Snell integrated the primary and secondary device sensors.

Assessment of Effort

Name	Signature	% of effort
Benjamin Sanati		19
George Palozzi		10.5
Przemyslaw Forys		10.5
Millicent Henley		8
James Wald		9
Kesavaram Krishnamoorthy		12

Gan Hock Nien		7.5
Henry Snell		16
Yevin De Costa		7.5

This is the workload breakdown, which was used to determine effort percent:

Ben: Server, bunch of research, HES, website GUI on my tree and our garden (navbar and background) pages, integration of hardware-server-website, server-website data handling, team leader
Harry: Simulation and design of analogue circuitry, physical build of devices, integration of sensor software including working with the sub teams on interfaces with other code. Filmed some clips for the video and did demo.
Przemek: Code for heart rate monitor, integrated all into single function
George: Research, design integrate power, Costing, Presentation (video, slides a bit + giving the presentation), Research (not used),
Milly: Website, JavaScript on it and design of some pages, small bit of initial database/server research
James: Made circuit from Harry's design, accelerometer + buttons for SD, not fully integrated with other modules of device
Yevin: Time, Timer, Database(only research), finalizing video advert
Hock: Timer, video editing
Kesavaram: Research for pedometer. Code for Pedometer and code for 85 percent of Timer inactivity code. In addition, pedometer and timer integration.