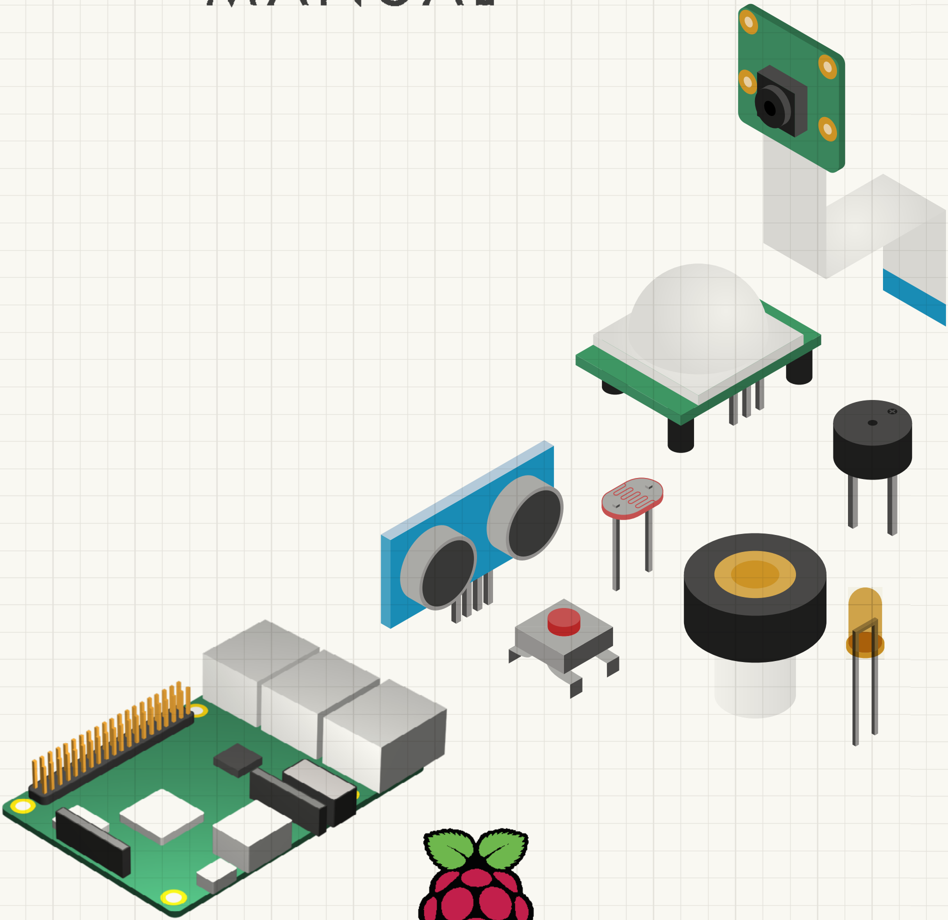
SOCIAL ACTION HACKATHON

DEVELOPER MANUAL



Raspberry Pi

SOCIAL ACTION USER/CLIENT QUESTIONS

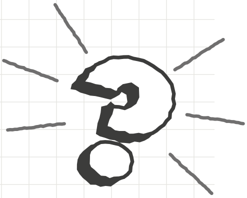
Identify three problems or difficulties your beneficiaries face, and write them down here:

1

2

3

SOCIAL ACTION USER/CLIENT QUESTIONS



Consider and discuss each issue independently using the following questions:

Your perspective:

- What are your thoughts and feelings regarding this situation?
- What are your assumptions about the situation?
- How might your own history and experiences influence your thoughts, feelings, and assumptions regarding the issue?

The beneficiary's perspective:

- From what perspective might your user(s) be viewing the situation?
- What appears to be most important to them?
- What aspect of the situation or problem do you think about the situation would elicit the strongest feelings or reactions for them, and why?
- What things might you have to take into consideration when providing solutions for them?

Creating solutions:

- What is the core of this issue? At the most basic level, what are you attempting to fix?
- Can you think of a technological solution to this issue?
- What would you require in order to build something that might help with this issue?

WHAT IS AGILE DEVELOPMENT?

The Agile development process can be summarised very briefly:

- Only work on the most important things at any given point in time
- Break those things into small bite-sized tasks for individuals to work on autonomously
- Catch up regularly on progress to work out what is important now, and change the plan to adapt

How does the process work?

- Your time at the hackathon is divided into small chunks of dedicated working called ‘sprints’. Each sprint is 90 minutes long.
- Everyone will have a specific role to play during each sprint, and you will swap roles every time. By the end of the hackathon, everyone will have done every job.
- At the very start, you will break your entire project down into individual jobs, and you’ll put these jobs on ‘tickets’. This pile of tickets is called the ‘backlog’, and it consists of everything that needs to be done before your project is finished. The backlog is displayed on your Kanban board – this is how you keep track of what has been done, and what still needs doing.
- At the beginning of each sprint, your startup team will meet and discuss whether you need to change the plan for the upcoming sprint, and to assess your progress. These meetings are called ‘stand-ups’, and they are really important!
- During the sprints, as you undertake each task in your backlog, move the ticket for that task from the ‘need to do’ column on your Kanban board to the ‘doing’ column. When you have finished the task, move the ticket to the ‘done’ column.
- Toward the end of your project, you will need to start thinking about your project pitch. We will have a special stand-up to discuss, plan, and create your pitch. You will organise your resources to put together a 1- to 2-minute presentation explaining your project, and your startup story, to the whole group and a panel of ‘Dragons’ at the very end.

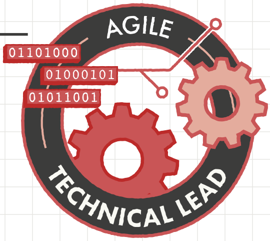
WHAT IS AGILE DEVELOPMENT?

What roles are we using in our Agile process?

COORDINATOR

- In charge of managing planning and workload requirements and external communications
- Responsible for chairing the startup at the beginning of a sprint
- Responsible for managing workload and resource allocation for the sprint
- Responsible for seeking outside help and communicating issues to facilitators
- Should lend a hand and get involved in making the project if the startup is experiencing time pressure or complications

TECHNICAL LEAD



- In charge of wiring, coding, and creating the internal workings of the device, according to the plan
- Responsible for bringing any technical issues or questions to the Coordinator
- Responsible for working closely with the Research Lead to find new solutions and work through the backlog of tasks

WHAT IS AGILE DEVELOPMENT?

RESEARCH LEAD

- In charge of finding new solutions and answers to technical questions using online searches and resources
- Responsible for helping Technical Lead implement complex solutions where necessary
- Responsible for liaising between Technical and Fabrication Leads to ensure that the interface/housing will work and fit the device
- Can be used as flexible workforce if the startup is experiencing time pressure or complications

FABRICATION LEAD

- In charge of creating the physical interface and the art-and-craft portions of the project
- Responsible for collecting and organising materials and components necessary for the project
- Responsible for working closely with the Research Lead to make sure that what they are building is fit for purpose

PRODUCT PLAUSIBILITY CHECKLIST

Some things aren't possible to achieve in just two days, on your first attempt, so make sure you aren't trying to do the impossible!

Just how possible is your idea? Does it require lasers, access to plutonium, or time travel? Can it actually be done? Discuss the following things within your startup to ascertain how plausible your project is before you commit to building it.

Timescale

You only have **about six hours** to design, plan, code, test, build, and implement your solution, so it should not be too complex.

- Aim for **around four to six features:** it takes roughly one hour to implement a single feature (having user input/control counts as a feature too). For example, a machine that lets you press a button to light an LED, take a picture, and then tweet it will take about four hours of work. So do you have enough time to do what you need?

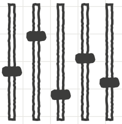
- Can/should you remove some features? Are there any that don't need to be there? What features are key to the design and shouldn't be removed if you do have to trim some?

Equipment

- We only have some basic components available, so we can't make flying, laser-guided robots!

- Are all the components (or access to WiFi/online services) you need for your design available to you? If you're not sure, ask!

PRODUCT PLAUSIBILITY CHECKLIST



Interface/controls

Your project is likely to be used by someone who doesn't know much tech, so you need to make sure it is easy to use.

- How would your product be applied by the user? Would it be in their home or on their person? Are there special considerations you will need to make for this?
- Will it need complex instructions to understand?
- Can you simplify it, or make it more user-friendly in any way?

Implementation

- How difficult will it be to make?
- What materials do you need? What is available? What can you bring in for the second day?
- Are there any restrictions or guidelines you need to be aware of that may affect the implementation of your solution (e.g. privacy, data protection, safeguarding, etc.)?
- Will your product require any special accounts or services to use? What backups can you provide if the user doesn't have these?
- Is your product designed to work with one specific service/app? Would the app provider be OK with what you're using their app for? Does it violate any of their terms of service?

PRODUCT PLANNING

Beneficiary and problem
Who are you helping? What problem are you solving for them?

Functions
How will your product solve this problem?
Write down all the functions your product will have.

PRODUCT PLANNING

Components
What parts will you need to achieve this functionality?
Write down all the components you will need to make your product.

Draw your product idea, labelling all the components you need.

PRODUCT PLANNING

Materials
What art supplies or craft materials will you need to create the casing or interface for your product?

Branding
What will you call your product? Write down a few names that you like here, and decide on one.

BACKLOG BREAKDOWN — USER STORIES

1

To get an idea of what your product should do exactly, tell yourself a simple story about how someone would interact with it and what would happen when they do. Write your user story down now, and make sure you cover all the things you want your product to be able to do!

2

Break this user story down into the required tasks to build your product, and write them all down.
Remember: For every task, you need to not only complete the job but also make sure that the functionality you're building works the way you want it to, **every time.**

BACKLOG BREAKDOWN — USER STORIES

3

Grab some post-it notes and start writing each job that needs doing on a separate note. Stick them all in the 'To do' column of your Kanban board.

For every **component** you add to your system, you need to add a ticket for:

- Collecting the component from the supply box
- Wiring it up to the Raspberry Pi
- Testing it with a simple script from the technical support documents to make sure it is working **before** you link it into the rest of your product with code
- Testing it once again **after** you have linked it to your system to make sure it's doing what you want it to!

For all the parts of the interface/case/stand you need to fabricate, you need to add several tickets. That includes tickets for jobs like:

- Collect required materials
- Get measurements from components
- Create simple design/blueprint for each part
- Check parts against product to make sure measurements are accurate and the fit is correct
- Making your product attractive and decorating it — that's a separate job as well!

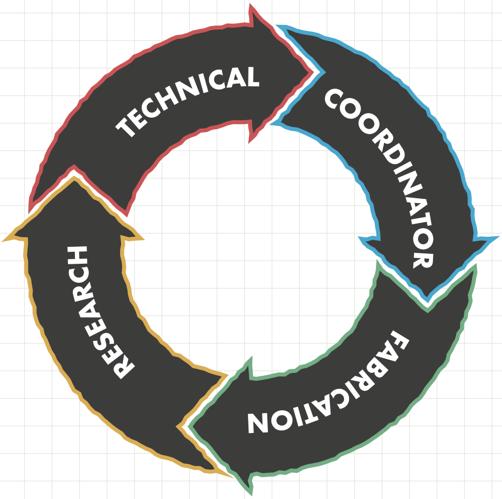
Sneaky tip: The more tickets you have, the better off you'll be when it comes to following the plan — no job is too small to have a ticket! And **always** make sure that you're moving each ticket when you begin doing the job, and again when you are done!

STAND-UP CHECKLIST

At the beginning of every sprint, your team must conduct a stand-up meeting to understand how you are progressing on your way to completion of your project.

Chaired by the person who is the Coordinator in this sprint (the person who was Technical Lead during the last sprint), stand-ups should answer specific questions for the Technical, Research and Fabrication Leads, so that the person undertaking each role in this sprint knows where they stand and what needs doing. It's about good communication, and not about pointing fingers about who did (or didn't do) what.

It doesn't matter which role you start with, but the order you will undertake the roles in is:



STAND-UP CHECKLIST

STEP 1

The person who was Coordinator in the last sprint should pass their lanyard to the person who was Technical Lead. They are now the new Coordinator!

New Coordinator:

Ask the person who **was** Fabrication Lead in the last sprint:

- What did you get done in the last sprint? How far along did you get with the backlog?
- What tickets do you need to focus on in this sprint?
- Are there any impediments in your way, or any things you found difficult that the new Fabrication Lead needs to know about?

Have the old Fabrication Lead hand over their lanyard to the person who was Coordinator during the last sprint.

STEP 2

New Coordinator:

Ask the person who **was** Research Lead in the last sprint:

- What did you get done in the last sprint? How far along did you get with the backlog?
- What tickets do you need to focus on in this sprint?
- Are there any impediments in your way, or any things you found difficult that the new Research Lead needs to know about?

Have the old Research Lead hand over their lanyard to the person who was Fabrication Lead during the last sprint.

STEP 3

New Coordinator:

Ask the person who **was** Technical Lead (this should be you!) in the last sprint:

- What did you get done in the last sprint? How far along did you get with the backlog?
- What tickets do you need to focus on in this sprint?
- Are there any impediments in your way, or any things you found difficult that the new Technical Lead needs to know about?

Have the old Technical Lead hand over their lanyard to the person who was Research Lead during the last sprint.

STAND-UP CHECKLIST

STEP 4

As a team:

Decide on what the Technical and Fabrication Leads need to focus on during this sprint, and on how your workforce and resources need to be allocated. Think about the following questions:

- What is your target at the end of this sprint? What stage should your product be at by then?

- Are you on track to finish your product, working at you current rate?

- Do you need to adjust the plan or change the design to make it simpler? How will you do that?

- What is the most important thing for the Technical Lead to get finished by the end of this sprint?

- What is the most important thing for the Fabrication Lead to get finished by the end of this sprint?

- What are the Research Lead and the Coordinator working on during this sprint?

    - Does anyone need extra help getting jobs done? Who will assist them?
    - You should never find yourself with nothing to do in a sprint! If you do, grab a ticket from the Kanban board and start helping!

STEP 5

Coordinator:

- If necessary, adjust the tickets on the Kanban board to reflect changes to your team's workflow (you may need to add some more or move some sticky notes to better reflect where you are).

- Make sure everyone knows their responsibilities for this sprint.

THEN GET GOING, EVERYONE!

THE PITCH
QUESTIONS TO ASK YOURSELF

At the end of the hackathon, your startup team is going to give a 2-minute presentation together to demonstrate your invention.

Your presentation should cover

1. Who is your social action beneficiary? What do they do?
2. What is the name of your product?
3. What problem does your product solve?
4. How does it work? How does the code work?
5. What did you use to make it?
6. How did you go about building it? Did you have any special considerations?
7. What problems did you face, or what unexpected things happened while you were developing your product? How did you overcome these? Did you have to change the plan at all?
8. How would you improve your product further if you had more time or resources?
9. Demonstrate how your product works to the group!

Some things to consider

How did you feel about digital making/coding before this hackathon?

How do you feel about it now?

How well do you think you communicated your views to your team?

How good was your team at listening to each other’s views and opinions?

Do you feel your understanding of others has increased?

How well do you feel you contributed to the team?

How well do you think the team worked together to find a solution,
and to create your product?

NOTES

NOTES

NOTES

NOTES

NOTES

Raspberry Pi