



University
of Glasgow | School of
Computing Science

Honours Individual Project Dissertation

MAKING PROJECTS: DYNAMIC PROJECT SYLLABUSES FOR PEOPLE WITH INTELLECTUAL DISABILITIES

Claire Williamson
April 11, 2023

Abstract

Inaccessible instructions and lack of knowledge of the needs of people with disabilities are two barriers preventing disabled people, especially people with intellectual disabilities, from participating in tangible learning. We designed and evaluated "Making Projects", a new software tool that adapts tangible learning project tutorials to suit users' needs by creating a customised syllabus of project tutorials. We explored the challenges people in this area of tangible learning have reported, and attempted to understand their impact on our projects and therefore how to build pathways through them to allow people with disabilities to develop strategies to overcome these challenges. We evaluated it one-on-one with three participants who had relevant experience with tangible learning and people with disabilities, in a lab study consisting of interviews and engagement with the tool. This revealed that participants found the tool intuitive, useful, and accessible for people with intellectual disabilities.

Education Use Consent

I hereby grant my permission for this project to be stored, distributed and shown to other University of Glasgow students and staff for educational purposes. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Signature: Claire Williamson Date: 24 March 2023

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 1.1 | Motivation | 1 |
| 1.2 | Aim | 2 |
| 1.3 | Objectives | 2 |
| 1.4 | User Story | 2 |
| 2 | Background | 3 |
| 2.1 | Making and Makerspaces | 3 |
| 2.2 | Designing for People with Disabilities | 4 |
| 2.2.1 | Social Model of Disability | 5 |
| 2.2.2 | Participatory Design | 5 |
| 2.2.3 | Universal Design | 5 |
| 2.2.4 | Ability-Based Design | 5 |
| 2.3 | Related Work: People with Disabilities Making | 6 |
| 2.4 | Related Tutorial Sites | 7 |
| 2.4.1 | Thingiverse | 8 |
| 2.4.2 | Make: Projects | 9 |
| 2.4.3 | Autodesk Instructables | 9 |
| 3 | Requirements Analysis | 11 |
| 3.1 | Stakeholders | 11 |
| 3.2 | Functional Requirements | 11 |
| 3.2.1 | Must Have | 11 |
| 3.2.2 | Should Have | 12 |
| 3.2.3 | Could Have | 12 |
| 3.3 | Non-Functional Requirements | 13 |
| 3.3.1 | Must Have | 13 |
| 3.3.2 | Should Have | 13 |
| 4 | Design | 14 |
| 4.1 | Overview of Making Projects | 14 |
| 4.2 | Syllabus of Project Tutorials | 14 |
| 4.2.1 | Syllabus Creation | 14 |
| 4.2.2 | Syllabus Design Considerations | 15 |
| 4.3 | Skill Area Requirements | 15 |
| 4.3.1 | Structure | 16 |
| 4.3.2 | Elicitation from User | 16 |
| 4.4 | Project Tutorials | 17 |
| 4.5 | User Account | 17 |
| 4.5.1 | Profiles | 18 |
| 5 | Implementation | 19 |
| 5.1 | Overview of Making Projects | 19 |
| 5.2 | Syllabus of Project Tutorials | 20 |
| 5.3 | Skill Area Requirements | 20 |

| | | |
|---------------------|--|-----------|
| 5.3.1 | Physical Tools | 21 |
| 5.4 | User Accounts | 21 |
| 5.5 | User Profiles | 21 |
| 5.5.1 | Selection of User Profile | 21 |
| 5.5.2 | Updating User Profile | 21 |
| 5.6 | Project Tutorials | 22 |
| 5.7 | General Site Appearance | 22 |
| 5.8 | Admin Interface | 22 |
| 5.9 | Site Hosting | 23 |
| 6 | Evaluation | 25 |
| 6.1 | User Evaluation Experimental Design | 25 |
| 6.1.1 | Participants | 25 |
| 6.1.2 | Pre-Study Interview | 25 |
| 6.1.3 | User Interaction with Making Projects | 26 |
| 6.1.4 | Post-Study Interview | 26 |
| 6.2 | Limitations | 26 |
| 6.3 | Analysis of User Evaluation | 27 |
| 6.3.1 | Removing Jargon | 27 |
| 6.3.2 | Legibility | 27 |
| 6.3.3 | Syllabus | 28 |
| 6.4 | Summary | 29 |
| 7 | Conclusion | 30 |
| 7.1 | Summary | 30 |
| 7.2 | Guidance for Similar Work | 30 |
| 7.3 | Future Work | 31 |
| 7.3.1 | Physical Projects | 31 |
| 7.3.2 | Optimisation of Syllabus Creation | 31 |
| 7.3.3 | Additional Features for Making Projects | 31 |
| 7.4 | Final Reflection | 32 |
| Appendices | | 33 |
| A Appendices | | 33 |
| A.1 | Ethics Approval | 34 |
| A.2 | Evaluation Consent Form | 36 |
| A.3 | Evaluation Introduction Script | 37 |
| A.4 | Information on Making Projects for User Evaluation | 38 |
| A.5 | Evaluation Debriefing Script | 39 |
| A.6 | Outline of Source Code | 40 |
| A.7 | User Manual | 40 |
| Bibliography | | 41 |

1 | Introduction

This chapter shall outline the motivations for making tangible learning more accessible for people with disabilities, and will present the aim of this project along with the related objective. We will also introduce a user story, which will be continued throughout as a way of putting this project into context.

1.1 Motivation

Tangible learning is simply the concept of learning by using physical objects. It has been found to be useful in teaching computer science-related concepts among others (Martin 2015), and can provide an effective and engaging alternative for education, especially for people whom traditional learning is not well suited for. Tangible learning can take many forms but of particular interest to this project are those that utilise technology. This includes electronics and robotics tool kits, digital fabrication, and tutorials for these accessible via the Internet. Technology can be used to enhance tangible learning, make it more accessible to a wider group of people, or carry it out in its entirety (Qi et al. 2018).

One group that could benefit from tangible learning are people with disabilities, specifically people with intellectual disabilities and other cognitive impairments. Intellectual disabilities are a group of neurodevelopmental disorders which greatly affect a person's mental ability, commonly affecting functions such as problem-solving, abstract thinking, and learning. These in turn cause impaired social and cognitive functioning, for example in memory, attention, and communication. (American Psychiatric Association 2013) In an increasingly digital world, disabled people on average have less digital literacy (Lloyds Bank 2022). Therefore it is important to find effective ways of educating - not just in teaching of computer science concepts, but by creating technology for carrying out that education. Tangible learning offers us a way to do so.

However, the current state of tangible learning remains inaccessible to many people with disabilities, especially people with reduced cognitive functioning (Seo 2019). Many factors can contribute to this, such as the equipment required being physically inaccessible. This could be due to them being specialised tools that average person does not own, or public workshops with these tools not being accessible for disabled people. There is also the issue of the tools requiring high levels of manual dexterity or vision, for example small buttons or screws. People with knowledge of working with people with disabilities often don't have knowledge of the tools and technology, and vice versa, leading to a knowledge gap (Rajapakse et al. 2015). Resources which could bridge this gap fall short - tutorials are frequently written with complex technical language, unclear and confusing instructions, assumptions about the person's ability, and in inaccessible formats.

Attempts have been made to lower these barriers and get people with disabilities involved with tangible learning. They have highlighted a need for adaptable resources that can be used in various situations, as well as expansion into including more complex projects to allow people to continue their learning journey. As people with disabilities gain more experience in tangible learning, they can develop strategies to overcome and address the constraints, challenges and difficulties they face whilst carrying out tangible learning projects.

This highlights the need of accessible tangible learning not just for introducing new concepts and learning opportunities for people with disabilities but also as a way to continue this learning in a way that suits them.

1.2 Aim

We propose Making Projects, a new tool which sorts tangible learning project tutorials dynamically based on an individual users needs. This is so that people with disabilities, specifically intellectual disabilities, can access the benefits of tangible learning. It aims to create a personalised syllabus of project tutorials so that the user can improve their knowledge and skills through a guided path.

1.3 Objectives

In order to achieve this aim, we have defined the following objectives:

- Create a process for capturing the requirements of the user in relation to tangible learning projects, in an accurate and accessible way.
- Design and implement an algorithm which dynamically creates a syllabus of project tutorials for an individual user.
- Design and implement a method and structure of displaying project tutorials in a way that is adaptable and accessible.
- Evaluate the tool with people who have knowledge of the area of tangible learning and people with disabilities.

1.4 User Story

In order to give a better understanding of the tool, we have created a user story which outlines a potential usage situation of Making Projects. We will return to it at relevant sections in order to illustrate what is occurring and why.

Ruby is a volunteer at her local disability support charity in Glasgow. She goes to their meetings twice a week for three hours, and spends time with a group of four people that are living with various intellectual disabilities. She spends this time trying to find activities to entertain and interest them. Ruby is keen to help the group to engage with science and technology more, but currently finds this difficult because she is a graphic designer and does not know much about science. She has tried to look up STEM activities online but has found all of them unsuitable for her group's needs. When they tried to follow one in the past, they found it was unsuitable halfway through and that they were not able to continue it. This led to her group feeling frustrated and not wanting to do any more of these activities.

With the Making Projects tool, she can describe the barriers each of the people she works with faces and get projects for them that they are able to do.

2 | Background

This chapter discusses relevant research which has formed the background of this project, firstly by looking at the current state of tangible learning and its interaction with people with disabilities. Then we will look at relevant design ideas for creating accessible objects, and prior work that involves bringing tangible learning to people with intellectual disabilities.

2.1 Making and Makerspaces

The preeminent form of tangible learning is the subculture of ‘making’, which has rapidly grown in both popularity and research interest. Making is centred around ‘makerspaces’ which are workshops accessible to the public. The number of makerspaces, or similar community workshops, in the UK has increased from only a few to almost a hundred in the past decade (Sleigh et al. 2015), showing that more people are getting involved with them as well as making them available to a wider range of potential users. These makerspaces allow for a sharing of tools and knowledge between ‘makers’ while they carry out their projects. Making is simply the process of physically creating something yourself. More than this, it’s become part of a global movement which champions this process over any end product. Kuznetsov and Paulos (2010) highlights the benefits “doing it yourself” brings: increasing skills with the tools used, understanding the technological and engineering concepts behind them, and the community aspect created from doing these projects alongside other people.

While traditional workshops are centred around woodworking and metalwork, makerspaces are notably different in their approach which focuses on technology, programming, and digital fabrication. This allows users to learn computing science concepts through the process of constructing and creating projects, as opposed to more traditional methods that require a lot of abstract thought and understanding (Martin 2015).

Tanenbaum et al. (2013) notes makerspaces as being sites of innovation, due to the fact that they allow more people the opportunity to design and quickly prototype new ideas. They allow more people from a variety of typically underrepresented demographics the opportunity to create their own products. This leads to a democratisation of product design, whereby the ability to design and make products is no longer solely limited to large companies. They allow people to create products that they personally want and need, and to easily modify the design to further suit them with the use of quick prototyping with digital fabrication tools.

One area that could benefit from this innovation is DIY-AT. The term Assistive Technology (AT) encompasses a wide range of devices that have the purpose of supporting people with disabilities and allowing them to function better. These could be items as simple as grips for cutlery, walkers, and grab rails, or high-tech devices such as hearing aids and motorised wheelchairs.

However, Phillips and Zhao (1993) finds that AT is regularly unused or abandoned by users. The reasons for this include:

- The user’s opinion has not been considered during the selection, leading to a device that is not suitable for their needs or that they don’t want to use.

- Devices, especially ones that are most effective for an individual, are difficult to obtain. This can be due to lengthy referral processes and/or the large financial cost.
- They do not perform well in the tasks required by the user.
- Disability and impairments are fluid in that they affect the person in different ways, often in a short span of time. This means that AT is no longer functional when the user's needs change, and the AT cannot adapt to this.

A few different ways have been proposed on how to improve continued AT use. One point is involving the end user more in both the design of technology, and the selection of it. One way this can be done is by “do-it-yourself” AT (DIY-AT). This allows the user more control over the design, resulting in AT that fits their personal needs. It can also be a more cost effective solution. This is important as traditional AT can be incredibly costly, especially when you take into consideration the fact that people with disabilities tend to earn less money (Office for National Statistics 2022), have higher unemployment rates (Department for Work & Pension 2022), and have additional expenses associated with having a disability (John et al. 2019).

Makerspaces offer an ideal environment for creating DIY-AT, which has been applied in practice. At the same time, there has tended to be a focus on non-disabled people designing and creating the AT which can lead to the same issues listed above. Hook et al. (2014) states that care must be taken to involve the person with the disability as much as possible in order to create a product that is of use to them.

Taylor et al. (2016) found that in addition to makerspaces being a site for peer learning and innovation, makerspaces function as places that support the well being of their users. They also have the ability to serve the needs of the community in which they are situated, provide spaces for socialisation, and work to reach out to groups that are excluded from such spaces. In the context of Making Projects, these features are particularly advantageous. People living with intellectual disabilities are more likely to feel lonely than people without the disability (Alexandra et al. 2018). Makerspaces therefore offer a space for them to connect with others and grow their social bonds.

Therefore, makerspaces seem to provide an ideal setting for tangible learning. Makers don't have the costly overhead of buying their own tools, and are surrounded by like-minded people willing to share their experience. These are factors which could greatly contribute to the breaking down of the aforementioned social and financial barriers that people with disabilities can face.

Despite this ideology of inclusivity and the motto of “anyone can make”, many makerspaces and making as a whole have issues with what groups are included. There is a disproportionate number of makerspace users that are affluent, non-disabled, white men with a STEM background (Maker Media, Inc. 2013). This lack of diversity by itself can discourage members of these underrepresented groups from accessing such spaces.

Many efforts have been made to lessen this disparity for various groups, to some success. Qi et al. (2018) looked at how their paper electronics toolkit engaged new groups and found that, similar to the e-textile community (Buechley and Hill 2010), paper electronics offers new ways to include groups in making. This shows that different forms of making can be utilised to effectively engage different demographics of people.

2.2 Designing for People with Disabilities

An important question raised for Making Projects is how to best design something for people with disabilities. In this section we will define and explain several concepts which are relevant to Making Projects.

2.2.1 Social Model of Disability

Central to how to design for people with disabilities is how we think about disabled people and the concept of disability. The social model of disability puts forward the idea that disability arises from situations where society has failed to meet the needs of such people, rather than from any limitations of an individual person (Oliver 1990). These barriers can be removed by designing in such a way that disabled people can have equal access, therefore making their disability negligible in this situation. It is in contrast to the medical model of disability, which instead considers disability an issue individual to the person and focuses on individual treatment and intervention. The emphasis is put on the medical implications, rather than the way the person with a disability interacts with society.

2.2.2 Participatory Design

Participatory design is centred around the concept of fully involving the end user in the design of the product (Persson et al. 2015) This allows the product to meet their needs and be usable by them. Many designs are inaccessible purely because the needs and requirements of disabled people have not been considered. It aims to combine the knowledge of the development team and the user, to create a better product. This concept ties in well with the culture of making, as a lot of the time the person creating the object has a lot of creative control over the outcome. It is also important for Making Projects, as a tool that is to be used by disabled people and so should be designed with their needs in mind to the greatest extent possible.

2.2.3 Universal Design

Arguably the most popular design term when designing for people with disabilities, as well as in general inclusive design. With the term first coined in 1985, universal design prioritises designing something which can be used by as many people as possible. This is further defined as “the design of products and environments to be usable by all people, to the greatest extent possible, without the need for adaptation or specialized design.” Center for Universal Design (1997) Its aim is to remove as many potential barriers as possible by taking a wider look at these issues, rather than just individually tackling each barrier. This links nicely to the social model of disability, in thinking about issues on a wider societal scale. In order to further explain this concept, the Center for Universal Design outlines seven principles:

- Equitable Use: The design is useful and marketable to people with diverse abilities.
- Flexibility in Use: The design accommodates a wide range of individual preferences and abilities.
- Simple and Intuitive Use: Use of the design is easy to understand, regardless of the user’s experience, knowledge, language skills or current concentration level.
- Perceptible Information: The design communicates necessary information effectively to the user, regardless of ambient conditions or the user’s sensory abilities.
- Tolerance for Error: The design minimises hazards and the adverse consequences of accidental or unintended actions.
- Low Physical Effort: The design can be used efficiently and comfortably and with a minimum of fatigue.
- Size and Space for Approach and Use: Appropriate size and space are provided for approach, reach, manipulation and use regardless of the user’s body size, posture or mobility.

2.2.4 Ability-Based Design

Universal design and similar approaches have their roots in designing physical objects and buildings to be accessible. However, computer software offers a much more flexible and adaptable

environment. Wobbrock et al. (2011) introduces ability-based design as a concept that builds off universal design to offer suggestions on how to use the unique properties of technology to design for disabled people. Its core principle is that there should be a focus on what the user can do, rather than what they cannot do. It presents 6 other principles that designers should follow when creating accessible technology, which we will briefly outline and discuss the ones relevant to our current work:

- Accountability: any issues with the system performance should be corrected by changing the system, not the user. This reflects the social model of disability in that it is a fault of the system that a user cannot access it, not a fault with the user.
- Adaptation: it's recommended that the system interface adapts to best fit each user's needs and abilities, either automatically or by the user. This results in a more accessible system, that is tailored to a users needs rather than a "one size fits all" approach which may inadvertently create issues.
- Transparency: the interface may tell the user what adaptations have been made, and allow them to modify them. It's important for the user to have control, as they best know what they need.
- Performance: the system may access the user's performance and use that data to measure, model, or predict their performance. This can allow for the system to better understand the needs of the user in order to better adapt to them.
- Commodity: it's encouraged that the system consists of low-cost and readily available hardware and software. This allows for more people to use the system, as there is no cost barrier.

2.3 Related Work: People with Disabilities Making

People with disabilities making combines the topics previously discussed. Prior work about people with disabilities, especially intellectual disabilities, carrying out tangible learning has revealed several relevant points. It has also highlighted where further research is needed. That is what we will discuss in this section.

Many making toolkits and resources are inaccessible to people with intellectual disabilities. This can be due to them requiring complex physical manipulation, relying on the user remembering prior concepts, and/or being in formats that are difficult to understand (Lefevre et al. 2016). In response to this, accessible tool kits have been developed to facilitate their tangible learning in a way suitable for their needs.

Ellis et al. (2021) developed a making toolkit "TapeBlocks" which incorporated electronics and simple motors. They aimed to lower the barriers for entry to making by creating an electronics toolkit that could function as an initial step into tangible learning/electronics. The core component of TapeBlocks are large foam blocks on which conductive tape is placed. These blocks can then be easily connected together to form different circuits. By using these large foam blocks, they lowered the level of fine motor control needed to create circuits, as well as not requiring a lot of physical strength as the foam is lightweight. The components are also affordable, and all these factors lead to their appeal.

Senaratne et al. (2022) then built on this work to create an accessible electronics toolkit for people with intellectual disabilities, that offered an increased amount of circuit creation activities. It aimed to lower the physical and cognitive barriers present in other electronics tool kits.

Several key themes have emerged from these works and other similar research:

Instructions:

Boccardi et al. (2022) found that it's best to not use technical jargon as well as breaking language down and keeping the projects simplified.

As well as avoiding complex technical language, writing instructions in Plain English can promote independent understanding. This is because people with cognitive impairments are more likely to understand these instructions on their own, without additional support. Having any instructions be available in different formats is important to allow for more people to access them (Brady et al. 2014). This could include having printed large text, digital, and braille format.

Project Considerations:

Brady et al. (2014) found that when deciding what types of projects to carry out or offer, consideration should be made of what different abilities the potential users will have. Any changes to the projects that they will need should be carried out in advance. If it is not possible for a project to be adapted, there should be another activity that they are able to do. Projects that are fully participatory for the user tend to be more successful in terms of interaction and enjoyment. Therefore creating projects that are suitable for someone to do as independently and as fully as possible, as in Making Projects, will improve the user's experience. Additionally, adults with intellectual disabilities preferred and benefited more from focused group activities. This suggests that carrying out projects in a group environment, such as a makerspace, is well suited to their needs. A point often echoed is that participants enjoy personalised projects and getting to create what they want. Boccardi et al. (2022) This helps foster a sense of independence and control in the participant. The reverse of this is also true, in that it is important for anyone supporting the disabled person to not take control of the situation, and instead allow them to problem solve.

Ellis et al. (2021) puts forward that projects themselves should aim to be affordable, require little barriers for engaging with them, and be able to be experimented with. This echoes similar principles in both universal and ability based design, emphasising the importance of their principles in such work.

Hollinworth et al. (2014) found that you can increase a person with intellectual disabilities understanding of how something works by having visual indicators. They modified the design of pre-existing electronics modules to make their functions clearer and easier to understand, as well as making each component look more different to each other. This allowed people with intellectual disabilities to more easily create electronic circuits, as well as having less errors during the process. This highlights the importance of visual design, especially when users may not be able to understand text.

Senaratne et al. (2022) suggests that making sessions should aim to be kept short (45-90 minutes) as keeping participants engaged and concentrated is difficult for longer periods of time. This can be achieved with Making Projects, as by having a range of project tutorials, a simpler one can be selected which will take less time to complete.

Ellis et al. (2021) states that the people who used their toolkit expressed an interest in learning and developing their electronics skills further, and wanted to know how to do that. Senaratne et al. (2022) did this partially by offering more ways to use electronics in an accessible way. This highlights a need for extending these beginner level tool kits to allow users to go further in their learning. However current research and resources are still limited in what extent this can be done, and only focusing on specific areas with tool kits.

Making Projects aims to build on the principles found in prior research to create a way for people with IDs to start and continue their tangible learning journey, to an open ended, more advanced level. By combining different areas of making projects, such as electronics and robotics, which have been developed through research.

2.4 Related Tutorial Sites

There are many online resources and websites that have the purpose of sharing tangible learning project tutorials. However, Wakkary et al. (2015) notes that often these tutorials are not of

high quality, leading to difficulties in implementing the projects. This can be due to issues with communication, sequencing of instructions, and the tools used. This section will provide a review of several of these resources, in order to identify their strengths as well as areas which may be inaccessible to people with intellectual disabilities. It will also discuss how they have influenced the development process and design of Making Projects, and how we aimed to address the areas of inaccessibility that are present in them. This process also served to understand what information our project tutorials should contain in order to be completed successfully by users. For each site, the example tutorial shown was selected by sorting the tutorials by "popularity" and choosing a suitable one.

2.4.1 Thingiverse

Thingiverse¹ is the largest online community for 3D printing. It is a website that contains over 2.5 million user-uploaded tutorials for 3D printing projects. It allows users to "remix" other people's tutorials, in order to customise them however they wish, and is aimed at people with all levels of experience with 3D printing. In theory, this is ideal for people with intellectual disabilities as there will be projects they can do as a complete beginner, and can then build up to more complex ones. In practice, Thingiverse does not have a way of denoting how complicated a project is, or filtering projects by their difficulty.

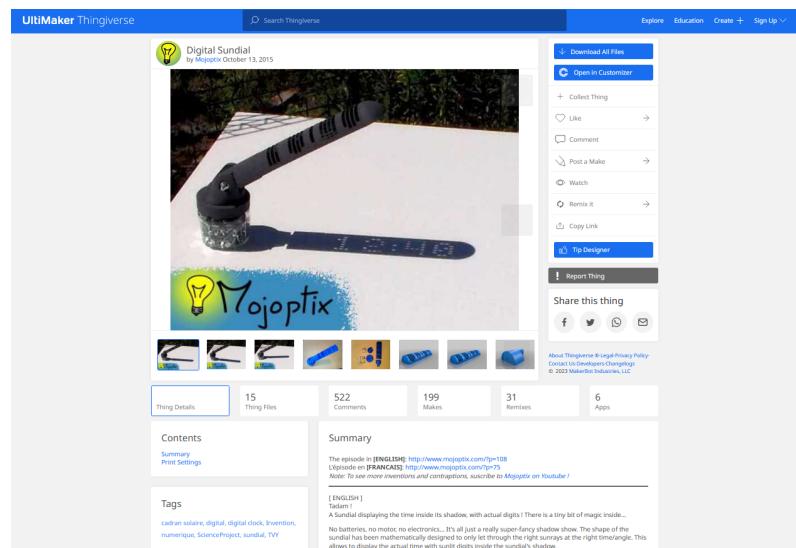


Figure 2.1: View of a tutorial for a 3D printed sundial on the Thingiverse website.

By its very nature, Thingiverse is limited in only showing projects that require a 3D printer. The information displayed for each project is relatively limited and gives no indication of what skills the user needs to be able to complete it. This is information that can be particularly important to those with intellectual disabilities to know prior to beginning a project, and its absence could dissuade such users from attempting the projects at all. Despite this lack of project information, the user interface contains a lot of different components which could be easily overwhelming and distracting for a person with intellectual disabilities (Chadwick et al. 2013).

¹<https://www.thingiverse.com/>

2.4.2 Make: Projects

Make: Projects² is a joint venture between Make:, a company focused around making which publishes the most popular magazine for the making community, and engineering.com, which is a collection of engineering based resources. Similar to Thingiverse, it functions as a space for users to upload and share their project tutorials. However, unlike Thingiverse these projects span across all types of making and DIY.

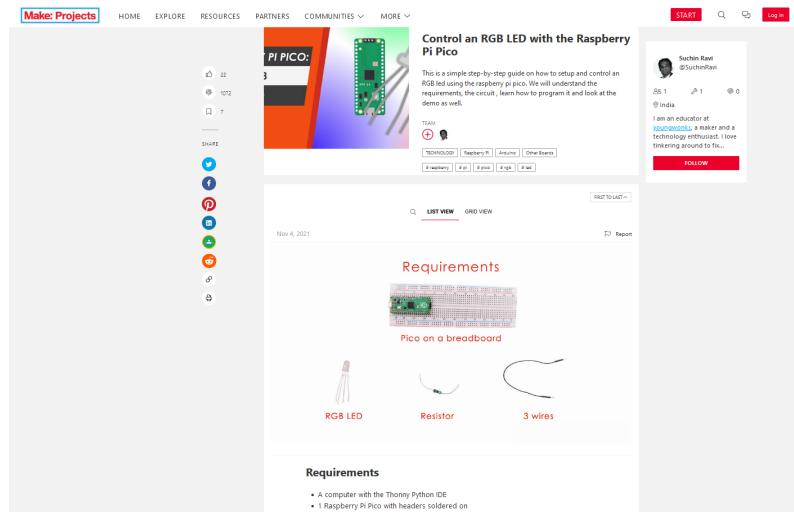


Figure 2.2: View of a tutorial for a LED controller on the Make: Projects website.

It has some identifiers for what a project contains, but these are limited to only the categories of projects. This could make the process of finding suitable and relevant projects a more time consuming task than necessary. Additionally the identifiers are in the form of small, similar looking icons which could cause confusion to a user. It shows less unnecessary information than Thingiverse, but still poses potential distractions.

2.4.3 Autodesk Instructables

Instructables³ is another site containing project tutorials. It is part of Autodesk which is a large software corporation with a focus on software for manufacturing. Similar to Make: Projects, Instructables attempts to cover a wide range of projects as possible.

²<https://makeprojects.com/home>

³<https://www.instructables.com>

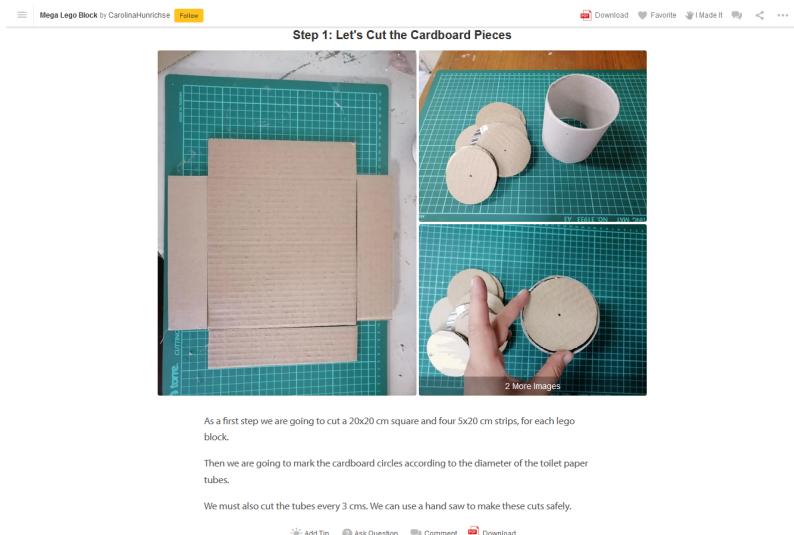


Figure 2.3: View of a tutorial for a cardboard Lego block on the Instructables website.

Out of the three resources discussed here, it has the most simplified user interface for the project tutorials. It focuses purely on the instructions and information needed to carry out the project. However, it has no filtering by or communication of project difficulty. Like with the other resources, the quality of the tutorials is dependant on the individual uploader which can lead to inconsistency. This can cause problems for all people attempting to complete this projects (Wakkary et al. 2015), but these problems will be further intensified for people with intellectual disabilities.

3 | Requirements Analysis

This chapter will discuss what Making Projects needs to have in order to fulfil its aims, in the form of its functional and non-functional requirements. These requirements were drawn from the points discussed in chapter 2, in addition to the following analysis of the project's stakeholders.

3.1 Stakeholders

From the above methods we identified various stakeholders and what their needs in this situation would be. This gave us a better idea of what the tool should do and how it should do it. The stakeholders we identified were:

- Target users: the people with disabilities themselves. They wish to learn more about tangible learning, and carry out projects that interest them. They may have a diverse range of impairments.
- Their supporters: people who would be coming into the space with the target user in order to assist them. These are people such as professional and family carers, and educators. They may not have any technical knowledge of tangible learning.
- Tangible learning experts: people who would already be in the space that have knowledge of making, and who are interested in helping others enter that space. These are people such as other makerspace users, instructors, and owners. They may not have much knowledge of working with people with disabilities and the challenges they would face in this situation.
- Disability advocates: professionals that would like to help more people with disabilities to engage with tangible learning. This includes disability support organisations, disability rights advocates, and medical professionals. They may wish to use the tool as a resource they can signpost people to.

3.2 Functional Requirements

These functional requirements detail specific behaviours and functions of the tool that will allow it to reach the aims discussed in chapter 1. These requirements are prioritised with the MoSCoW method, which gives an indication of how important a feature is to the final product achieving its goals. The requirements are also given itemised codes and titles, in order to be able to easily refer to them throughout the project. These itemised codes also reflect the level of priority the requirement has, and that it is a functional requirement.

3.2.1 Must Have

These are features that the tool needs to have in order to meet the aims given in chapter 1.

FR-MH-1 Project Tutorials: The tool must allow users to view various making project tutorials. The tool adapts the tutorials it shows in order to meet the user's needs. Users can clearly see if a project meets their requirements. The user does not have to log in or create an account to view

them. This will allow people with disabilities to get project tutorials that are accessible to them, in order to carry them out and gain the benefits of tangible learning.

FR-MH-2 User Accounts: Users can easily register for an account with the tool.

FR-MH-3 Information on Skill Areas: Logged in users can tell the tool what their current skill level in various areas is. The areas will be relevant to the types of projects, as well as common barriers people with intellectual disabilities face in tangible learning. Users should be able to clearly tell what effect their information has, and can easily change it.

FR-MH-4 Physical Tools Used for Making: Users can input what physical tools they have access to and their current experience level with that tool. Similarly, the project tutorials will state what physical tools are required for that project and the level of experience needed.

FR-MH-5 Syllabus: Once a user has logged in and given their information on skill areas and physical tools, the tool must allow them to create a syllabus. The user specifies a project tutorial that they want to be able to do, and the tool will create a path of projects which the user can follow to gain the skills needed for their selected project.

Ruby works with four people: Cody, Rex, Kira and Deanna. Deanna has problems with her physical coordination, and has poor eyesight, which makes doing tasks such as screwing small electronic components very difficult for her. However, through the process of working through a Making Projects syllabus, she works out ways to make this easier. One way she does this is by placing any components on a white background and using a magnifying glass so that she can more clearly see them. This allows her to work through the projects, perhaps at a slower pace, but she can still complete it.

3.2.2 Should Have

These requirements are not essential but are important aspects in creating a functional tool.

FR-SH-6 Record of Project Completion: logged in users can indicate when they have completed a specific project. If the project is in their syllabus, then a progress bar can be displayed. The user's profile can display what projects they have completed.

3.2.3 Could Have

The following requirements are not needed to meet the aims of Making Projects but would improve the user's experience with it. These are also potential areas of future work, as the current project has a limited time frame.

FR-CH-7 Guide for Project Adaptation: the tool contains a guide on how to adapt project tutorials to make them more accessible for people with disabilities. This could then be used by people with existing tutorials in order to make them more accessible. It could also be used alongside the creation of new tutorials to ensure that they are suitable.

FR-CH-8 Dynamic Project Tutorials: Having a single project being adaptable so that users with different requirements can complete it.

FR-CH-9 DIY-AT Project Tutorials: the tool contains project tutorials that involve creating a piece of DIY Assistive Technology, which could allow the person to more easily complete other making projects as well as improving their general day-to-day functioning.

3.3 Non-Functional Requirements

These non-functional requirements specify general properties of the tool. As with the functional requirements, they are prioritised using the MoSCoW method and itemised accordingly.

3.3.1 Must Have

These are aspects which the tool must have in order for it to meet its aims.

NR-MH-10 Ability Based Design: The tool must follow the principles of ability-based design as discussed in chapter 2. It must focus on what a user can do, rather than what they cannot do.

NR-MH-11 Universal Design: The tool is able to be effectively used by a wide range of people as possible, and follows universal design guidelines where feasible. This includes considerations such as making content accessible to screen readers, and using clear language.

NR-MH-12 Freedom of Choice: Users have freedom to choose what projects they would like to do.

NR-MH-13 Extensibility: The tool is created in such a way that it is easy to add new components, without having to alter pre-existing code.

NR-MH-14 Maintainability: The tool's code base, as well as the tool as a whole, is designed and well documented so that others can readily continue working on it.

3.3.2 Should Have

These are aspects which would greatly improve the design of the tool, but are not necessary for a successful project.

NR-SH-15 Embedded Tutorials: Project tutorials should be in a simple format so that they can easily be embedded in other sites.

NR-SH-16 Participatory Design: The design and development of the tool is based on the real experience of the intended end users. The end users should be involved to the highest level possible.

4 | Design

This chapter will discuss the final design of the tool, beginning with an overview and then looking at specific key elements. It will detail important design choices, along with their justification and the relevant requirements they aim to meet.

Making Projects underwent a major refactor halfway through. Important changes from which are highlighted in the relevant sections in both the design and implementation chapters. This refactor was due to the project specification evolving as we got a better idea of what the tool should do, and allowed for components to be more effectively designed and more clearly separated.

4.1 Overview of Making Projects

The Making Projects tool will consist of a library of making project tutorials that can be adapted to an individuals specific needs. As well as these individual projects, the tool can create a customised syllabus of projects for a user, in order to create a path towards their specified goal. The user will tell the tool their needs, which it will use to make the adaptions resulting in projects and syllabuses which that user is able to complete.

The system structure for this can be described using the Model-View-Controller architecture. In which the user makes requests through the view (the user interface), which then sends this request to the controller to determine the appropriate action. In some cases this action requires the controller to query specific data from the model (in our situation, this is a database). It then processes this data to return a response to the user.

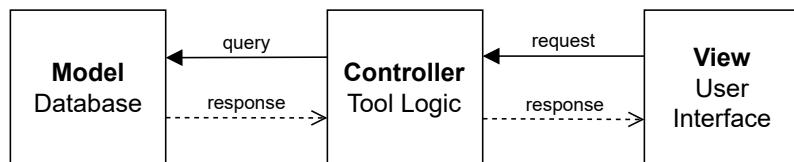


Figure 4.1: Model-View-Controller Architecture Diagram.

4.2 Syllabus of Project Tutorials

The main feature of the tool is the automated syllabus creation. Its purpose is to allow the user to build up their skills in manageable steps so that they are able to complete a project of their choosing.

4.2.1 Syllabus Creation

The design of the syllabus creation is as follows.

```

while end project requirements > profile requirements do
    Find what skills to improve;
    Find a project with an improved skill;
    Add that project to syllabus;
    Update the profile's requirements;
end

```

Firstly, the user interface asks the user to choose what project they want to be able to do. It sends this request to the controller, which queries the database for the information on that project. This information includes specifics regarding what skill levels are needed to successfully carry out this project, as well as the physical tools.

Making Projects then queries the database for the user's information, which will contain information in a similar structure to that of the project tutorial. It compares the skill level information of the project with that of the user. If the user already has suitable skills to carry out this project, then the syllabus will only contain the chosen project.

In most cases, the user won't already have the skills needed. In that situation, the controller calculates what areas they need improve. Then it goes through each of these areas, and attempts to find a suitable project tutorial. This is a project which has identical skill area levels to that of the user, apart from in the specified area for improvement whose level will be increased by one. If no project tutorial can be found matching those requirements, it repeats this process with the next skill area and so on. If no project is found for any of these skill areas needing improvement, it repeats the search process for each area but with the conditions that all other skill area's can be equal to or less than the user's current profile values. This increases the chance that a complete syllabus can be created, without needing projects with every combination of skill requirement values.

Once a project is found, it is added to the syllabus. The user's skill area requirements are then updated to reflect this project i.e. the value of one skill area will increase by one. Then this entire process will repeat until the user has all the skill levels needed to carry out their end project. At that point, the end project will be added to the syllabus, and the syllabus as a whole will be presented to the user.

4.2.2 Syllabus Design Considerations

The original syllabus design was that there would only be three or four set syllabuses that a user could choose from. This idea was changed so that the end project choice could be completely customised by the user. This is important in meeting the requirement **NR-MH-12**, in allowing the user to make their own choices. It also allows for more flexibility of the system, as with the final design it is still possible to have set syllabuses but not the other way around.

Another design consideration is that of the granularity of the skill levels. In other words, the jump in difficulty between each value, and the total range of values for a skill. We decided on an initial range of 1-3 for each skill area, which is discussed further in section 4.3. If the jump between difficulty levels was found to be too great, greater granularity could be added, or multiple projects with that same difficulty could be added to the syllabus before one with that increased level.

4.3 Skill Area Requirements

Another key area which is involved in a many areas of the tool is that of the skill area requirements. These are common skills present in tangible learning projects, and which people with disabilities struggle with. They are used to categorise the needs of both users and projects in regards to these

areas.

It is important for the tool to have a way of accurately gauging both what a user can do, and what is needed on behalf of the user to complete a project.

4.3.1 Structure

This information was further split up into abstract skills, and physical tools used for projects.

The skill areas we chose to focus on are drawn from the relevant literature as ones which people with intellectual disabilities and other cognitive impairments have issues with, especially in the context of tangible learning. We decided to select four of these areas for our initial tool, but additional area's can be easily added or changed. The areas which we selected are as follows:

Dexterity: The physical motor skills needed for tangible learning projects can be a pivotal barrier. People with intellectual disabilities commonly have motor impairments. Clearly stating what level of manual dexterity is needed to complete a project will allow both the tool and the user to gauge if they can complete a project.

Language: As discussed, issues with instructions are prominent. As well as having all instructions be clear, having more complex language in instructions allows for more advanced projects. There is a need for flagging what complexity of language is used in a project. Additionally, the use of images alongside text can be used by people who may not be able to understand written language, as well as improving the understanding of those who do.

Memory: Many tutorials make assumptions about the users cognitive load, which is how many concepts a person can remember. People with intellectual disabilities typically have reduced functioning in this area. For example, project tutorials that signal that they are suitable for people with a lower cognitive load may explain relevant concepts frequently before a user needs to apply them.

Vision: A persons level of vision can have a large affect on how they can access tutorials and the actual physical process of making. Therefore it is important to make clear how difficult a project is in this area.

4.3.2 Elicitation from User

The process of getting these requirements from the user is important. Accurately understanding the needs of a user is difficult for both the tool and the person. To make this process as simple and effective as possible we designed a form for the user to select what their needs are. We wished for this form to focus on what the user was able to do (in keeping with **NR-MH-10**), and so it contains a series of statements that explain what a person can do and the corresponding skill level for that ability. It was also crucial to have clear examples for each skill level, rather than abstract concepts that could be difficult to understand. In following with the ability-based design principles (Wobbrock et al. 2011), we also needed the impact of these requirements to be clear to the user, as well as being easily updated or modified by them.

Ruby's group's are all affected by their intellectual disabilities in different ways, and to different extents. When Ruby wants to find an activity that they can do, she has to consider each person's individual needs which can get confusing. Often she ends up with activities that her group are not really interested in doing. With Making Projects, she can support each person in figuring out what are they able to do, which they can then use to see if projects are suitable for them. For example, she can sit down with Cody and work out that since he has poor physical coordination he has the equivalent "low" dexterity level on Making Projects. Likewise, he does not have impaired vision so his vision level is "high", and so on for each skill area.

4.4 Project Tutorials

The content of the projects themselves will mostly be taken from various online sites, covered under Creative Commons licence, and manually adapted by us to follow these guidelines which are drawn from chapter 2.

- All written text is as near as possible to being in plain English. Writing in plain English is a learned skill which requires experience, and we were unable to dedicate the time to doing so. However it provided a logical point to aspire to when writing text for the project.
- Instructions consist of clear numbered steps. This allows users to easily see the order of what they have to do, as well as helping in the troubleshooting process by outside parties.
- The format was easy for screen readers to process. While we aimed for the entire tool to be suitable for screen reader technology, a particular focus was placed on the project tutorials as the main feature of the tool.

We also had to process what skill area levels were appropriate for each project. The types of projects selected focus on various technological concepts such as digital fabrication, electronics, and robotics. They will also cover various levels of complexity in regards to their skill area requirements. This will allow a larger proportion of users to have project tutorials they are able to do. The actual project tutorials themselves were not the main focus of this project, and so they may not be completely accurate. Instead, focus was put on the general structure of the project tutorials and their interaction with the tool.

The project tutorials interact with the users information to display if they are at a suitable level. Users do not have to be logged in to view projects, as we wish for them to be accessed by the widest range of people and situations.

4.5 User Account

Users can register and log in to the tool. This allows for a users data to persist over several sessions, so that they can gain the benefits from continued use. This also removes the extra time taken to go through the process of entering skill area requirements every time – which could require considerable effort for a person with an intellectual disability. Additionally it removes the burden from the user of remembering their skill area requirement values. We wished for the information that needed to be remembered by the user to be as minimal as possible, and in a format where it is likely to be recalled. Therefore we decided to use an email address to log in over a username. Our rationale was that as an email address is an already existing piece of information it is more likely to be recalled over a new username. However there is a counterpoint to this in that an username is typically shorter, and people repeat usernames over different platforms. We decided that this point could be discussed in future study with actual participants.

4.5.1 Profiles

We needed a way in which to store a users skill area requirements and other relevant information. Originally we contained all this information within the user account. However during the refactor, we decided to separate them out into two objects. This allowed each user account to have multiple profiles, improving the flexibility of the system, as well as containing all information about a user profile in one place. One particular use case that was considered was that of an instructor working with multiple people with disabilities. This is common in settings such as specialised disability support services, or with researcher-led projects. Therefore this change allowed an instructor to easily switch between the profiles of each person they were working with. Additionally, this makes it easier for future development to use the functionality of user profiles without needing a user account.

5 | Implementation

This chapter describes how we implemented our design, with a discussion of what software was used and the reasoning behind it as well as specific implementation details.

5.1 Overview of Making Projects

The tool was implemented using Django¹, which is a popular Python web framework. It contains a built in development server, which allows the project to be quickly deployed on the local machine. This was useful as it automatically updates changes made to the code to the display, allowing for a faster development process.

Additionally, Django aims to follow don't repeat yourself (DRY) principles in the structure of its applications. This made it easier for the project to contain separate components, and clearly define the interactions within them.

The core of our tool was a number of database models with related methods, as seen in 5.1. These held all the data related to the skill area requirements, users, user profiles, and projects. Then, we created functions to manipulate this data in response to user requests, which would be returned using our web page template code.

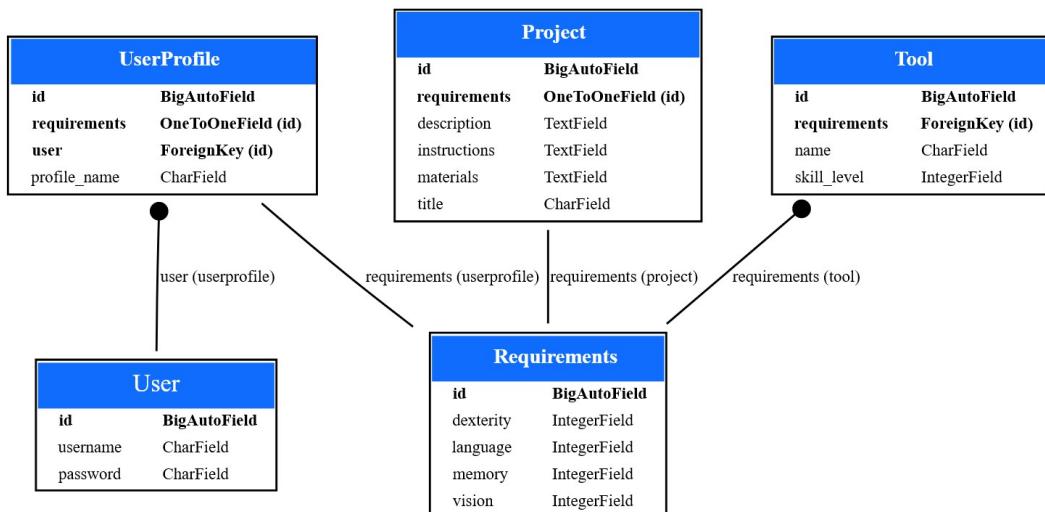


Figure 5.1: Diagram of the Django models in Making Projects. One-to-many relationships are signified with a black circle.

¹<https://www.djangoproject.com/>

5.2 Syllabus of Project Tutorials

The syllabus creation algorithm is the core feature of the tool. It was implemented as a Django view function, which is Python code that is run when the user interacts with the syllabus creation page. It was redesigned several times in order to find a solution that worked well. We will outline the final implementation, along with the justification for several features which were shaped by our refactoring process.

The algorithm consisted of several separate functions, in order to make it clearer and reuse code. Firstly it requests the data it needs about the "end goal" project (end project). This calls a custom function of the Project model in order to only get the fields that are required. This function also collects all the related objects of the project, such as the skill area requirements and any physical tools that it uses. Originally this was done separately in the view function, but was redesigned to repeat less code and make more effective use of the database. An equivalent model function for the User Profile object is then called.

Then it enters a while loop, which each time checks if the users profile is equivalent or greater than the end project, in terms of skill area levels. If they are not, it then calls a function that determines what areas the user still needs to improve which are returned as a list.

This list is then passed into another function, along with the user's profile, which iterates through the list and queries the database for a project that matches the user profiles current values with that skill level incremented by one. Projects that contain tools that the user doesn't have access too, or not enough experience in, are not selected. If a project is not found on this first pass of the list, it reiterates through it but this time searches for projects with the incremented skill value, but accepts all other values that are equal to or less than the current user profile. At first this constraint relaxing was not a feature, but was added in order to increase the chances of a syllabus being produced. We decided to keep it in conjunction with the first search function, as it would be more beneficial to a user to complete a project that better matches their requirements, if possible.

Once a project is found, its information is appended to the syllabus, and the copy of the users profile is incremented accordingly. Then the while loop is ran again until the condition is met.

Once this happens, the view function checks if the end project is in the syllabus and if not, appends it to the end. Then it calls the related model function on each project in the syllabus in order to get the information needed to display it for the user. This information is passed to the renderer, so that the user can see their syllabus.

5.3 Skill Area Requirements

The skill area requirements were implemented as a model, with the structure shown in figure 5.1. Originally skill area requirements were implemented as fields within their respective project or user profile objects. However we realised that these fields were repeated, and instead redesigned it as a separate model. This allowed for methods to be written that used this model, rather than having to repeat it for both the user profile and project models. This also allows for the possibility of having multiple requirements objects per project, which could then be used to adapt an individual project to a users needs as in requirement **FR-CH-8**. This was not implemented in this version of Making Projects, due to time constraints.

Each skill area has three choices, which are the same for each skill but unique choices for each skill can easily be implemented without causing any loss of functionality.

5.3.1 Physical Tools

The information about the physical tools were implemented as their own model. The ID of the related requirements object was stored as a field. This allowed for a one-to-many relationship between requirements and tools. Having them as separate objects made it simpler and more flexible when comparing the skill area requirements of user profiles and project tutorials. Users can only add tools to their profiles which are present in any of the project tutorials, this was implemented by creating a custom model manager that returns all distinct tool names present in the database and sets these values as the potential choices. Similarly, there is a constraint which prevents multiple tools with the same names being associated with one requirements object. They are four choices in what level of experience they have with that tool. Tool instances are deleted when their related requirements object is deleted.

5.4 User Accounts

The user accounts were implemented using Django's built in authentication system. This allowed us to utilise other features such as password hashing and validation functions. We chose to modify it to use the least amount of data as possible, in the form of a username and password. We opted to use an email instead of a traditional password, on the basis that it was more likely to be remembered. Despite it appearing as username in the code, due to Django requiring this, all forms were adapted to validate for an email address.

5.5 User Profiles

We separated out user accounts and user profiles, as detailed in chapter 4. User profiles were implemented as a Django data model, which is then mapped to a relational database by Django's ORM. The user profile's requirements are stored as a separate data model, and linked to the profile.

5.5.1 Selection of User Profile

Since it is possible for one user account to have multiple profiles, the tool had to be able to decide which profile to use at various points. To make it as consistent as possible, the profile which is to be used is stored as a session variable. When a user logs in, if they only have one profile then it is automatically selected. If they have no profiles, the session variable is set to the value "None", and they are redirected to the "Create Profile" page. When a user has multiple profiles, their session variable is set to the value "None" and they are redirected to the "Select Profile" page so they can choose which profile they wish to use. This page can also be used to switch which profile is the session variable at any point during the session.

When a user creates a new profile, their session variable is set to that new profile. Session variables are deleted when the user logs out or when the browser is closed. This option can be toggled, and is currently set to delete on browser close to increase storage space and overall efficiency. However there is the consideration that it would be helpful for the end user to have their profile remain selected.

5.5.2 Updating User Profile

The user can update their details on the site. The page's template fetches their current details and displays them as the default values in the form so the user knows what their current values are.

5.6 Project Tutorials

Similar in nature to the user profile, project tutorials were implemented as data models with related requirements models. All the functions surrounding them were stored within the data model, so that they were all in one place and could easily be used without rewriting code.

It was important for the project tutorials to contain images. However instead of containing an absolute path to the images in the data model for each project, we created a function which dynamically looked up the file path based on a projects ID value and returned the images found in that directory.

5.7 General Site Appearance

The appearance of the tool was a crucial consideration. It needed to be appealing as well as functional for people with many different abilities. We decided to use Bootstrap², which is a front-end website toolkit, for the site's appearance. This allowed us to have an attractive looking website with minimal time spent on writing CSS. It also contains many of the best practices for web design, including some which we added, such as having hidden links for screen reader software to skip content, clearly naming all links, and using accurate alternative text for images.

Django website pages are created using templates, which provided an ideal way to define a singular page design which could then be extended for each page leading to a consistent site design without repeating any code. The final design of the site can be seen in 5.2.

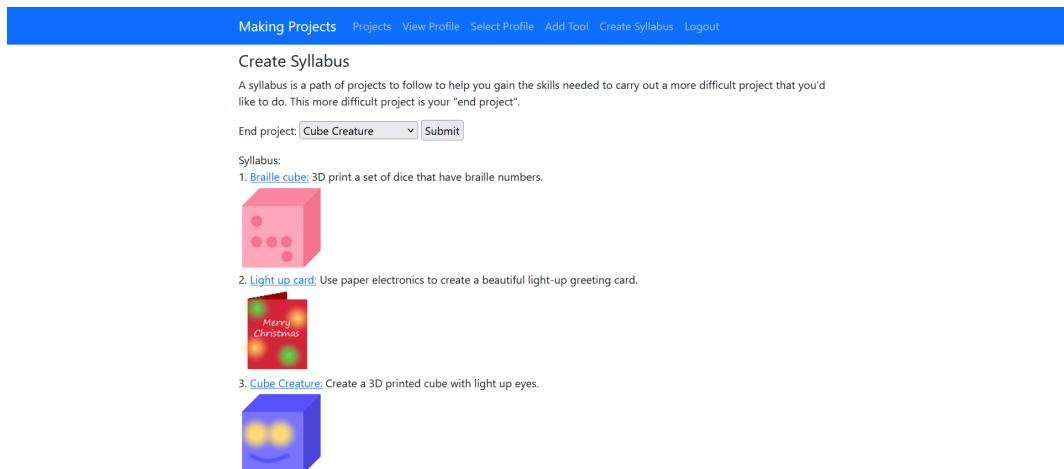


Figure 5.2: View of the syllabus page of Making Tools, showing the user interface design.

5.8 Admin Interface

Django comes with a built in admin interface, which allows an admin user to interact with the database models through the site. We modified this to better suit the structure of our models and add additional functionality. As our tool used many related objects, one modification we made was displaying links to these objects when viewing said related object, as seen in 5.3.

We also added inline fields, which allowed a models related objects to be viewed and edited in the same place as the model. This can be seen in 5.4. This makes it easier to understand and work with the related data, while still allowing the actual models to be separate.

²<https://getbootstrap.com/>

| ID | TITLE | DESCRIPTION | MATERIALS | INSTRUCTIONS | REQUIREMENTS |
|----|--------------------|---|--|--|--------------|
| 30 | Paper Card Machine | Twist the pencil, and one card will appear and the other disappear. | standard envelope, pencil, 3-4 rubber bands,card | <ul style="list-style-type: none"> Seal the empty envelope line up a playing card with its short end along the envelopes short side Cut envelope about 2cm past the card Turn the envelope so the open end faces up. About $\frac{1}{4}$" below the open end, cut or tear small half circles on either side Decorate envelope Twist and wrap 3-4 rubber bands around the middle of your pencil Carefully push pencil through the holes Now, you can put a card on either side of the pencil in the envelope and twist the pencil to reveal each one | 30 |
| 1 | Braille cube | 3D print a set of dice that have braille numbers. | printer filament, sandpaper | <ul style="list-style-type: none"> Download the dice files by right clicking here. Load the file into your 3D printer Load up the filament Set it to print the dice. Sand printed dice if needed. | 1 |
| 3 | Cube Creature | Create a 3D printed cube with light up eyes. | LEDs, sandpaper, printer filament, tape | <ul style="list-style-type: none"> Download the creature file. Load file into printer Load filament into printer Print the file Sand edges of finished creature if needed. Put LEDs through the eye holes Tape the wiring to the inside of the creature | 3 |
| 2 | Light up card | Use paper electronics to create a beautiful light-up greeting card. | card, LEDs, tape | <ul style="list-style-type: none"> Create a design on the card Create holes in the card with the scissors for the LEDs Thread the LEDs through the holes Secure their wires with the tape Present the card | 2 |

Figure 5.3: Image showing the admin view of project tutorials, displaying the links to their related requirements object.

| NAME | SKILL LEVEL | DELETE? |
|------------|-------------|--------------------------|
| computer | Expert | <input type="checkbox"/> |
| 3d printer | Expert | <input type="checkbox"/> |
| circuits | Expert | <input type="checkbox"/> |
| | None | <input type="checkbox"/> |
| | None | <input type="checkbox"/> |
| | None | <input type="checkbox"/> |

Figure 5.4: Image showing the admin view of a requirements object, which shows its related tool objects and allows modification of them.

This was done to allow future developers, or other people using the tool, to easily modify the data without having to interact with the actual code. All of these changes were collected in one file that clearly labels the changes made.

Now that Ruby has carried out a few projects with members of her group, she has a greater understanding of how the projects should be. She has found a project online that she would like to do with Cody, but would prefer to have it on Making Projects as he is used to using it. Since her organisation uses its own hosted version of Making Projects, she can log into it as an admin and easily add this new project, despite not having any knowledge of programming.

5.9 Site Hosting

The Django framework provides an ideal structure for web apps that can be easily hosted in different places. For this project, Making Projects was hosted using PythonAnywhere³ which is a free to use Python development environment that allows hosting of Python projects. It was selected as it is low-cost and well suited for hosting Django projects, both in its capabilities and its resources on how to host a Django web app.

However, Making Projects itself can readily be hosted on a number of platforms. The code base con-

³<https://eu.pythonanywhere.com/>

tains clear instructions on how to do so, informed by the process of hosting on PythonAnywhere. This includes how to generate a secret key, and how to store it securely.

6 | Evaluation

This chapter discusses the evaluation methods used for the tool, the results from those methods and their relation to the overall project requirements. We analyse these results in order to get an effective evaluation of Making Projects.

Making Projects is designed to be used by people and so a user evaluation was important to understand how well the tool serves its purpose.

6.1 User Evaluation Experimental Design

The user evaluation was run as a lab study due to it being an initial evaluation of the tool. We also did not have the time or resources to carry out an evaluation in a makerspace or disability support group. This is discussed in more detail within 6.2 limitations.

Further evaluations outwith a lab study can benefit from the findings of this initial user evaluation. The results from this evaluation can also improve Making Projects before any more complex studies.

The evaluation consisted of participants interacting with Making Projects with the experimenter observing. It conformed with every requirement from the University of Glasgow School of Computing Science ethics checklist, as shown in Appendix A.

Participants were asked permission for an audio recording of the entire evaluation. The consent form can be seen at Appendix A. After written consent was given (no participant withheld permission), the audio recording was stored on a secure device, transcribed and rendered anonymous, and then all copies of the recording were deleted.

One evaluation was carried out in person, and the other two through video calls and screen sharing.

6.1.1 Participants

The participants for the study all had knowledge of tangible learning and people with disabilities. We had approached participants for their knowledge in either area, but as it turned out they all had various degrees of experience in both. We classified knowledge of tangible learning as any involvement with makerspaces or similar organisations, or individual experience with hands on learning. Knowledge of people with disabilities included personal experience of having a disability, as well as working in roles that supported disabled people.

6.1.2 Pre-Study Interview

Before each participant interacted with Making Projects, they had a semi-structured interview with the experimenter. The use of this structure allowed participants to share any information they thought relevant, while still covering all the same points in each interview. This interview lasted around 20 minutes and aimed to get an understanding of the participants experience with tangible learning, makerspaces, and people with disabilities. We also gathered demographic information in the form of their gender identity and age.

Table 6.1: Overview of each participants demographics and experience in various areas.

| ID | Gender | Age | Making | Makerspaces | Disability | |
|----|--------|-------|----------|-------------|------------|--------------|
| | | | | | Personal | Professional |
| P1 | F | 18-25 | High | Moderate | High | None |
| P2 | F | 18-25 | Moderate | None | High | Moderate |
| P3 | F | 18-25 | High | High | High | Moderate |

6.1.3 User Interaction with Making Projects

Following the pre-study interview, participants were given access to Making Tools via a link to the hosted site on PythonAnywhere. This allowed participants to utilise any personalised adaptations they have made to their device, which is likely to occur in actual usage situations. This also allowed the site to be tested on multiple different types of device/screen. Participants were told to consider a potential user for Making Projects, and with that person in mind, carry out certain tasks. This was in order to evaluate the tool in relation to the intended end user, rather than the participant. The tasks were:

- view a project tutorial
- register for an account
- enter their skill area requirements, and physical tools
- select a suitable end goal project
- create a syllabus with that project

The observation allowed us to see how the participant navigated the site, and any issues they may have had.

6.1.4 Post-Study Interview

After using the tool, there was another interview with the participant. This was to gather their opinions of the tool. They were asked to identify any specific accessibility issues they noticed while using the tool, with the tool in general and also with the presentation of the project tutorials. Then they were asked which, if any, part of the tool they thought would be most useful in lowering the barriers to allow people with disabilities to carry out these projects. They were also invited to give any other opinions of the tool.

6.2 Limitations

There were several limitations present in our user evaluation, and therefore our evaluation of Making Projects as a whole.

While Making Projects is intended to be used by people with intellectual disabilities, involving these people in such a project requires a college level ethics approval application. This is a timely and complex process, with no guarantee of the application being approved, so the decision was made to not go ahead with it due to our limited time frame. However this did impact on our ability to fulfil requirement **NR-SH-16** as we could not interact with the intended end users.

Instead, we reached out to local makerspaces and organisations which support people with intellectual disabilities. We introduced our project to them and asked if they would be willing to give their feedback on it and take part in an evaluation. We received encouraging responses from a few organisations, however when we attempted to arrange a follow up for an evaluation no organisations responded in time.

Despite this, we remain hopeful that these organisations are amenable to be re-approached for future work in this area.

In order to still carry out a meaningful evaluation of Making Projects, we then approached the people discussed in section 6.1.1. This was not our ideal situation, but at this point our time frame was incredibly limited.

6.3 Analysis of User Evaluation

Three main themes emerged from our analysis of the user evaluation results:

Removing Jargon How the tool removed the need for knowledge of technical language, as well as areas where this could be improved.

Legibility The way in which the tool itself was accessible, and aspects which were not.

Syllabus How the syllabus worked and its perceived benefits to a user.

We will discuss each of them in more detail, and give an overall view of our findings from the evaluation.

6.3.1 Removing Jargon

All participants, to different extents, highlighted the use of plain non-technical language as beneficial to a potential user. They believed that this would lessen the amount of technical knowledge needed by a user to complete the projects.

Specific areas were identified as doing so, such as the explanations for skill levels and what each value meant. This was deemed particularly helpful to a potential user, as it effectively communicated what each option meant in a way which they could understand. One participant suggested that our choice of lesser known examples could hinder this, such as Duplo (a type of Lego block which is larger and easier to use), and that to aid understanding the tool could contain images of the objects given in the examples. We believe this to be a good idea, and have noted it in our areas for future work.

One participant with less knowledge of tangible learning found the project tutorial instructions simple and easy to follow. They suggested that any acronyms be spelt out before being used, even ones we may consider basic or unneeded such as LEDs, in order to make the information more accessible to people with intellectual disabilities as well as screen reader technology.

Another participant noted that displaying the skill level values as plain numbers across the tool was confusing as the context did not make it clear as to their meaning, and added an additional layer of complexity. Instead they suggested that we use relevant labels for the levels, rather than the raw numbers which is how they are stored in the database. We then implemented this, as these labels already existed for user input form and merely had not been extended across the tool. This allowed for a more cohesive and understandable design. We also made sure we were not using any similar numbers in other places of the tool, and instead using informative descriptors.

6.3.2 Legibility

A crucial point was that of having the tool be accessible and usable, in the context of people with intellectual disabilities. Participants views on this were shaped by their interactions with the tool as well as the points brought up by them in the interviews.

Overall, the simple design of the user interface was described as beneficial by the participants as it made clear what a user could do and avoided confusion. However, all participants pointed out that the contrast between the navigation bar's background colour and its text was not high enough. This led to the text fading into the background slightly, which was noted as being a

potential problem as it removed focus from the navigation bar. Additionally it could be difficult to read for people with visual impairments or issues with reading text. This can be changed in a future iteration of Making Projects to allow for a more effective and user friendly design.

Another point raised was the significance of clearly showing the user the results of their actions. This was successful in some areas, such as the automatic redirection when a user created an account, and first line of text on the page visibly changing when a user switched profiles. One place where this was not successfully done however, was when the user created a new profile. Multiple participants noted that the current design did not make it obvious if the profile had been made or not, as the text indicating so blended in with the rest of the information. One participant took a while longer to realise it had been successful, whereas another participant did not see this text at all and instead created another profile. This problem can be addressed by putting more emphasis on any such text, and improving tolerance for error by giving users feedback on what has occurred in a way that they readily understand so they can respond accordingly.

The use of external technology was also a discussion topic that came up in our interviews. It is likely that an intended end user has custom modifications to their technology which they wish to use when using Making Projects. These modifications aim to adapt the person's experience of using the technology in a way that is most accessible for their individual needs. Therefore it was important to the participants that both the external technology and Making Projects continue to function correctly when used together. This external technology could include software such as screen readers and display overlays, or hardware such as differently sized screens, and uncommon input devices. Two participants asked if the tool would work well with screen reader technology, and highlighted the importance of all links on the site being labelled correctly.

While we had made a point to design the tool in a way that could be used by as many people as possible, we had not intended to comprehensively evaluate the effects of different external technology on its functioning. However we were encouraged to discover that Making Projects could be used alongside one participants modifications, which included a "dark mode" plugin that altered the colours of the site, and a text highlighter and reader.

6.3.3 Syllabus

All participants were in agreement that the syllabus was the most helpful feature of the tool. They found the site's explanations of the syllabus and how to interact with it were intuitive, and did not require any further explanation from the experimenter.

One participant discussed the importance of the syllabus, in that they believed it would allow a user to achieve their end goal. They had had the prior experience of wanting to be able to do something but not knowing how to get there, and were delighted by the prospect of being given a comprehensible list of projects.

"If you're on your own and you know you want to achieve something but you're not sure how to get there, it will show you the intermediate steps."

Another point that was brought up was how the syllabus consisted of multiple, clearly defined "steps" (the projects). Participants agreed that this structure was beneficial in improving a persons knowledge, and allowing them to develop strategies for tangible learning. One participant added to this by sharing their lived experience of doing similar projects with a disability, and how a common pitfall was that they tried to do too much in too little time. They believed that this structure would help prevent this happening, by giving a user manageable tasks to complete separately.

6.4 Summary

The user evaluation revealed problem areas of the tool which could then be addressed and improved. It also highlighted the importance of considering individual users needs, especially when designing for people with disabilities. As well as allowing us to gain a better understanding of how people would interact with the tool. We were satisfied that Making Projects met all its “must-have” functional and non-functional requirements.

7 | Conclusion

In this chapter we will provide a summary of the entire project, along with our guidance for similar work, areas for future work that we have identified during this process, and a final reflection on this project as a whole.

7.1 Summary

Tangible learning offers a fun alternative to traditional teaching methods. It has successfully been utilised by the maker movement to improve physical and abstract skills, create a community, and improve people's well being. However despite these benefits being well suited for people living with disabilities, especially intellectual disabilities, tangible learning as a whole remains inaccessible. Barriers such as physically demanding equipment and complex instructions prevent them from fully engaging with tangible learning.

In order to facilitate tangible learning for people with disabilities, we proposed Making Projects, a tool which adapts project tutorials in a way that makes them easier to complete, and creates a customised syllabus of project tutorials.

Making Projects uses the Django Python framework to create a modular and extensible web app that can leverage a users' needs to find a suitable course of projects. It can also calculate if a specific project is currently suitable for a user.

We evaluated Making Projects with an user evaluation and found that it removed the need for complex technical language, was usable by people with different disabilities, and was effective in creating a syllabus of project tutorials. Combined, these findings shows that the tool is successful in lowering some of the barriers currently present in tangible learning.

A hosted version of Making Projects can be found at ciw.eu.pythonanywhere.com.

7.2 Guidance for Similar Work

Along with the areas outlined in section 7.3, the process of doing this project has revealed some aspects we believe that similar work could find helpful.

Firstly, there should be a focus throughout the entirety of the project on involving the end user as much as possible. If this is not feasible, as in our case, then attempts should be made to involve people with experience of the end users. This will allow for an end product that is more suited for its purpose and therefore more efficient.

When creating something to be used by people with disabilities, it should allow for as much customisation as possible. While we were aware of this from researching Universal Design (Center for Universal Design 1997), it was not made apparent until the user evaluation the sheer degree of customisation that can be beneficial to a user.

7.3 Future Work

During the course of this project we have identified potential areas for future work, both to improve Making Projects, and for the general research field of improving people with disabilities access to tangible learning. Some of these areas are requirements from chapter 3 that we did not have enough time to fulfil.

7.3.1 Physical Projects

Making Projects focused more on creating an environment to facilitate tangible learning, rather than the tangible learning projects themselves. Therefore future work on what tangible learning projects are most suitable, and how to overcome common accessibility issues with them, is vital in allowing more people with disabilities to access tangible learning.

DIY-AT In addition to this, project tutorials that result in a piece of assistive technology (AT) was a requirement of this project (**FR-CH-9**) which we did not have the time to complete. However it is still a relevant area which could provide great benefit for people with disabilities, both in their tangible learning processes and day to day life.

7.3.2 Optimisation of Syllabus Creation

In its current form, the syllabus creation algorithm is sufficient. However it is inefficient as in the worst case, it requires multiple searches of all the projects in the database. This will be exacerbated as more projects are added to the tool. Therefore it would be beneficial to optimise this process. This could be done by sorting the database, or refactoring the algorithm to require less searches.

Changes could also be made with the order in which skill areas are chosen for improvement. At the moment, skill areas are improved first if a suitable project exists, before any physical tools. This feature could be modified in conjunction with user evaluation of the specific projects in a syllabus, to result in a syllabus that is most effective in teaching a user.

7.3.3 Additional Features for Making Projects

There are many features that could improve Making Projects, several of which are discussed below. These were either requirements that did not have time to be implemented, or which were made apparent during the user evaluation of the tool.

Record of project completion Allowing users to see what projects they have previously completed was a requirement (**FR-SH-6**) that did not manage to get implemented. It was independently brought up by participants during the evaluation as a good potential feature to have. Making Projects having this would allow users to feel a sense of achievement, as well as not requiring users to recall what projects they have already done. This is relevant as people with intellectual disabilities often have impeded memory.

Project Tutorial Filtering Being able to find a project that you want, and are able, to do is a process integral to Making Projects. However the users experience of this could be further improved, as made apparent during the user evaluation. Features such as a search engine that returns projects with a certain skill level, for example low dexterity, could allow users to see their options for project tutorials. An indication of the skill level needed on the tutorial previews could also improve this. This could be in the form of small simple icons representing each skill, with the number of icons representing how much experience in that skill is needed.

7.4 Final Reflection

This project has taught me a lot about how to manage and organise a large piece of work, especially its importance when the overall time frame is limited. I also learnt how crucial it is to follow best practices when designing a software system, particularly when considering modularity. It became even more clear after my understanding of the project developed and I attempted to add new features to my original design, only to have to redesign the entire structure. In retrospect, using a database engine that allowed for greater filtering of queries such as PostgreSQL would have allowed for greater extensibility for features such as a search engine for project tutorials. SQLite is lightweight but at the cost of having less functionality. The process of evaluating this project has allowed me to understand just how important user evaluations are for a tool like this, in order to gain awareness of how effective the tool is in its intended environment. Finally, the process of writing this paper has taught me valuable skills in how to effectively convey the purpose and development of my project through academic writing.

A | Appendices

A.1 Ethics Approval

**School of Computing Science
University of Glasgow**

Ethics checklist form for 3rd/4th/5th year, and taught MSc projects

This form is only applicable for projects that use other people ('participants') for the collection of information, typically in getting comments about a system or a system design, getting information about how a system could be used, or evaluating a working system.

If no other people have been involved in the collection of information, then you do not need to complete this form.

If your evaluation does not comply with any one or more of the points below, please contact the Chair of the School of Computing Science Ethics Committee (mattthew.chalmers@glasgow.ac.uk) for advice.

If your evaluation does comply with all the points below, please sign this form and submit it with your project.

1. Participants were not exposed to any risks greater than those encountered in their normal working life.
Investigators have a responsibility to protect participants from physical and mental harm during the investigation. The risk of harm must be no greater than in ordinary life. Areas of potential risk that require ethical approval include, but are not limited to, investigations that occur outside usual laboratory areas, or that require participant mobility (e.g. walking, running, use of public transport), unusual or repetitive activity or movement, that use sensory deprivation (e.g. ear plugs or blindfolds), bright or flashing lights, loud or disorienting noises, smell, taste, vibration, or force feedback
2. The experimental materials were paper-based, or comprised software running on standard hardware.
Participants should not be exposed to any risks associated with the use of non-standard equipment: anything other than pen-and-paper, standard PCs, laptops, iPads, mobile phones and common hand-held devices is considered non-standard.
3. All participants explicitly stated that they agreed to take part, and that their data could be used in the project.
If the results of the evaluation are likely to be used beyond the term of the project (for example, the software is to be deployed, or the data is to be published), then signed consent is necessary. A separate consent form should be signed by each participant.
Otherwise, verbal consent is sufficient, and should be explicitly requested in the introductory script.
4. No incentives were offered to the participants.
The payment of participants must not be used to induce them to risk harm beyond that which they risk without payment in their normal lifestyle.
5. No information about the evaluation or materials was intentionally withheld from the participants.
Withholding information or misleading participants is unacceptable if participants are likely to object or show unease when debriefed.
6. No participant was under the age of 16.

Parental consent is required for participants under the age of 16.

7. No participant has an impairment that may limit their understanding or communication.
Additional consent is required for participants with impairments.
8. Neither I nor my supervisor is in a position of authority or influence over any of the participants.
A position of authority or influence over any participant must not be allowed to pressurise participants to take part in, or remain in, any experiment.
9. All participants were informed that they could withdraw at any time.
All participants have the right to withdraw at any time during the investigation. They should be told this in the introductory script.
10. All participants have been informed of my contact details.
All participants must be able to contact the investigator after the investigation. They should be given the details of both student and module co-ordinator or supervisor as part of the debriefing.
11. The evaluation was discussed with all the participants at the end of the session, and all participants had the opportunity to ask questions.
The student must provide the participants with sufficient information in the debriefing to enable them to understand the nature of the investigation. In cases where remote participants may withdraw from the experiment early and it is not possible to debrief them, the fact that doing so will result in their not being debriefed should be mentioned in the introductory text.
12. All the data collected from the participants is stored in an anonymous form.
All participant data (hard-copy and soft-copy) should be stored securely, and in anonymous form.

Project title _____ Dynamic Making Projects_____

Student's Name _____ Claire Williamson_____

Student Number _____ 2464406_____

Student's Signature _____  _____

Supervisor's Signature _____

Date _____ 14/12/2022_____

A.2 Evaluation Consent Form

**School of Computing Science
University of Glasgow**

Consent Form
Dynamic Making Projects
Level 4 Dissertation Project
Claire Williamson

I have been briefed about the evaluation I am about to participate in and I understand what is involved with it. The evaluation will be audio recorded so it can be used to take notes of the interview and will be destroyed after the Level 4 Individual Project deadline (**Friday 31st March 2023**). Data will be stored securely and in an anonymised form.

By signing this form I confirm that I have been briefed and I give my consent for the interviewer to audio record this interview.

Date: _____

Participant's signature: _____

A.3 Evaluation Introduction Script

**School of Computing Science
University of Glasgow**

Evaluation Introduction

Dynamic Making Projects

Level 4 Dissertation Project

Claire Williamson

The aim of this experiment is to evaluate the design and implementation of the “Making Projects” tool. This tool aims to make finding and carrying out Making project tutorials more accessible to people with disabilities. It is in the form of a website that displays projects based on a user’s inputted requirements. ‘Making’ projects combine physical objects with simple electronics and software - they can involve everything from 3D printing, electronics and prototyping boards to using cardboard.

This evaluation should take around 30-45 minutes.

Firstly I will ask you some questions to gauge your experience and first impressions with this topic. I will then give you a tour of the tool and give you some time to browse it. After that I will give you some tasks to complete with the tool, before asking you some questions about what you thought of it. I will observe you performing the tasks, and record your responses to the questions.

With your consent, this interview will be audio recorded so that it can be used to take notes. This data will be destroyed after the Level 4 Project Deadline (**Friday 31st March 2023**). If you do not consent to this, written notes will be taken during the interview instead.

All of this data will be stored securely, in an anonymous form. Please ask questions if you need to at any point.

Please remember that it is the tool, not yourself, that is being evaluated.
You are welcome to withdraw from the evaluation at any time.

Contact details: Claire Williamson 24644064w@student.gla.ac.uk
Dissertation supervisor: Dr Stephen Lindsay stephen.lindsay@glasgow.ac.uk

A.4 Information on Making Projects for User Evaluation

**School of Computing Science
University of Glasgow**

Making Projects Information

'Making' is simply the process of physically creating something yourself. More than this, it's become part of a global subculture which champions this process and the benefits "doing it yourself" brings - from increasing skills with the tools used, to understanding the technological and engineering concepts behind them, and the community aspect created from doing these projects alongside other people in a "makerspace". This culture is centred around makerspaces, which are workshops open to the public that allow for sharing of both tools and the knowledge of other participants.

Despite this ideology of inclusivity, many makerspaces and "making" as a whole remain inaccessible to people with disabilities. Reasons for this include makerspaces simply being physically inaccessible, and tutorials being in formats disabled people cannot understand or use. One less tangible barrier is people with knowledge of Making not having knowledge of people with disabilities and how to work with them. And then, people that have knowledge of disabled people (i.e. carers, disability coaches) not having knowledge of Making. Both these groups' knowledge is required in allowing these disabled people to get the benefits of Making.

"Making Projects" aims to tackle these barriers. It is essentially a library of 'maker' project ideas that are adapted to an individual's specific needs. It attempts to bridge this knowledge gap to allow people with disabilities to get the benefits from 'making'. As well as individual projects, users can create a customised "syllabus" of projects to help build their skills toward their specified goal.

"Making Projects" is designed with a focus on people with intellectual disabilities, brain injuries and cognitive impairments, but also intends to be helpful to people with other disabilities too.

Please note that the projects currently on the website are not "real", in that they aren't expected to actually be carried out. We are not trying to evaluate the actual projects, rather the structure and design of them. We apologise if this makes it harder to evaluate the site. Please do ask any questions at any point.

A.5 Evaluation Debriefing Script

**School of Computing Science
University of Glasgow**

Debriefing Information

Dynamic Making Projects

Level 4 Dissertation Project

Claire Williamson

The aim of this experiment was to evaluate the design and implementation of the “Making Projects” tool. I wished to find out how you interacted with the tool, and your opinions on its accessibility. The data that I collected with your permission, is an audio recording of the evaluation. This recording will be stored securely, transcribed and made anonymous, and then deleted no later than the project submission deadline (**Friday 31st March 2023**).

Please ask any questions, or give any comments about the experiment.

Please take a note of my email address and the email address of my dissertation supervisor, and let us know if you have any further questions about this experiment.

Thank you very much for your help.

Contact details: Claire Williamson 24644064w@student.gla.ac.uk

Dissertation supervisor: Dr Stephen Lindsay stephen.lindsay@glasgow.ac.uk

A.6 Outline of Source Code

Individual Project - Claire Williamson

Dynamic Project Syllabuses for People with Intellectual Disabilities

This repository contains the code, time log, supervisor meeting minutes, status report, and evaluation information for my individual dissertation project.

Making Projects is a tool which adapts tangible learning project tutorials in a way that makes them easier to complete by people with intellectual disabilities, and creates a customised syllabus of project tutorials. It is implemented using the Django Python framework.

Making Projects is currently hosted at: ciw.eu.pythonanywhere.com/

Directory structure:

- `data/` all data collected during the project, including all items related to ethics
- `dissertation/` final project dissertation pdf and its source materials
- `meetings/` meeting minutes from supervisor meetings
- `presentation/` slides and video for the project presentation
- `src/` source code for Making Projects
- `status_report/` the December status report pdf and source material
- `timelog.md` the time log for the project
- `plan.md` week-by-week plan for the project

For more detailed information, view that directory's readme file.

A.7 User Manual

Readme

Making Projects Django project.

Build Instructions

Requirements

- Python 3.8
- Django 4.1.6
- SQLite 3.41.1
- Packages listed in `requirements.txt`
- Tested on Windows 11

Build steps

It is advised to use a virtual environment.

Clone repository to local machine, cd to src directory. Activate virtual environment, if you are using one.

Using pip:

```
pip install -r requirements.txt
```

Set up secret key:

Create a file named `.env` in the src directory.

Within it:

```
export SECRET_KEY='key_here'
```

where `key_here` is your secret key (usually a random combination of symbols)

Then to run site on local machine (127.0.0.1:8000/):

```
python manage.py runserver
```

Bibliography

- Alexandra, P., Angela, H. and Ali, A. (2018), 'Loneliness in people with intellectual and developmental disorders across the lifespan: A systematic review of prevalence and interventions', *Journal of Applied Research in Intellectual Disabilities* 31(5), 643–658.
- American Psychiatric Association (2013), *Diagnostic and Statistical Manual of Mental Disorders*, fifth edn, American Psychiatric Publishing.
- Boccardi, A., Szucs, K. A., Ebuenyi, I. D. and Mhatre, A. (2022), 'Assistive Technology Makerspaces Promote Capability of Adults with Intellectual and Developmental Disabilities', *Societies* 12(6), 155.
- Brady, T., Salas, C., Nuriddin, A., Rodgers, W. and Subramaniam, M. (2014), 'MakeAbility: Creating Accessible Makerspace Events in a Public Library', *Public Library Quarterly* 33(4), 330–347.
- Buechley, L. and Hill, B. M. (2010), LilyPad in the wild: How hardware's long tail is supporting new engineering and design communities, in 'Proceedings of the 8th ACM Conference on Designing Interactive Systems', DIS '10, Association for Computing Machinery, New York, NY, USA, pp. 199–207.
- Center for Universal Design (1997), 'Center for Universal Design', <https://design.ncsu.edu/research/center-for-universal-design/>.
- Chadwick, D., Wesson, C. and Fullwood, C. (2013), 'Internet Access by People with Intellectual Disabilities: Inequalities and Opportunities', *Future Internet* 5(3), 376–397.
- Department for Work & Pension (2022), 'The employment of disabled people 2021', <https://www.gov.uk/government/statistics/the-employment-of-disabled-people-2021/the-employment-of-disabled-people-2021>.
- Ellis, K., Dao, E., Smith, O., Lindsay, S. and Olivier, P. (2021), TapeBlocks: A Making Toolkit for People Living with Intellectual Disabilities, in 'Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems', CHI '21, Association for Computing Machinery, New York, NY, USA, pp. 1–12.
- Hollinworth, N., Hwang, F., Allen, K., Kwiatkowska, G. M. and Minnion, A. (2014), Making electronics more accessible to people with learning disabilities, in 'CHI '14 Extended Abstracts on Human Factors in Computing Systems', CHI EA '14, Association for Computing Machinery, New York, NY, USA, pp. 1255–1260.
- Hook, J., Verbaan, S., Durrant, A., Olivier, P. and Wright, P. (2014), A study of the challenges related to DIY assistive technology in the context of children with disabilities, in 'Proceedings of the 2014 Conference on Designing Interactive Systems', DIS '14, Association for Computing Machinery, New York, NY, USA, pp. 597–606.
- John, E., Thomas, G. and Touchet, A. (2019), Disability Price Tag Policy Report, Technical report, Scope.

- Kuznetsov, S. and Paulos, E. (2010), Rise of the expert amateur: DIY projects, communities, and cultures, in 'Proceedings of the 6th Nordic Conference on Human–Computer Interaction: Extending Boundaries', NordiCHI '10, Association for Computing Machinery, New York, NY, USA, pp. 295–304.
- Lefevre, K., Totzauer, S., Bischof, A., Kurze, A., Storz, M., Ullmann, L. and Berger, A. (2016), Loaded Dice: Exploring the Design Space of Connected Devices with Blind and Visually Impaired People, in 'Proceedings of the 9th Nordic Conference on Human–Computer Interaction', NordiCHI '16, Association for Computing Machinery, New York, NY, USA, pp. 1–10.
- Lloyds Bank (2022), 'UK Consumer Digital Index 2022', <https://www.lloydsbank.com/banking-with-us/whats-happening/consumer-digital-index.html>.
- Maker Media, Inc. (2013), An in-depth profile of makers at the forefront of hardware innovation., Technical report, Make Community LLC.
- Martin, L. (2015), 'The Promise of the Maker Movement for Education', *Journal of Pre-College Engineering Education Research (J-PEER)* 5(1).
- Office for National Statistics (2022), Disability pay gaps in the UK 2021. www.ons.gov.uk/peoplepopulationandcommunity/healthandsocialcare/disability/articles/disabilitypaygapsintheuk/2021.
- Oliver, M. (1990), The Individual and Social Models of Disability, in 'PEOPLE WITH ESTABLISHED LOCOMOTOR DISABILITIES IN HOSPITALS', pp. 1–7.
- Persson, H., Åhman, H., Yngling, A. A. and Gulliksen, J. (2015), 'Universal design, inclusive design, accessible design, design for all: Different concepts—one goal? On the concept of accessibility—historical, methodological and philosophical aspects', *Univ Access Inf Soc* 14(4), 505–526.
- Phillips, B. and Zhao, H. (1993), 'Predictors of Assistive Technology Abandonment', *Assistive Technology* 5(1), 36–45.
- Qi, J., Buechley, L., Huang, A. b., Ng, P., Cross, S. and Paradiso, J. A. (2018), Chibitronics in the Wild: Engaging New Communities in Creating Technology with Paper Electronics, in 'Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems', CHI '18, Association for Computing Machinery, New York, NY, USA, pp. 1–11.
- Rajapakse, R., Brereton, M., Sitbon, L. and Roe, P. (2015), A Collaborative Approach to Design Individualized Technologies with People with a Disability, in 'Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction', OzCHI '15, Association for Computing Machinery, New York, NY, USA, pp. 29–33.
- Senaratne, H., Ananthanarayan, S. and Ellis, K. (2022), TronicBoards: An Accessible Electronics Toolkit for People with Intellectual Disabilities, in 'Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems', CHI '22, Association for Computing Machinery, New York, NY, USA, pp. 1–15.
- Seo, J. (2019), 'Is the Maker Movement Inclusive of ANYONE?: Three Accessibility Considerations to Invite Blind Makers to the Making World', *TechTrends* 63(5), 514–520.
- Sleigh, A., Stokes, K. and Stewart, H. (2015), 'Open dataset of UK makerspaces: A user's guide', <https://www.nesta.org.uk/report/open-dataset-of-uk-makerspaces-a-users-guide/>.

- Tanenbaum, T. J., Williams, A. M., Desjardins, A. and Tanenbaum, K. (2013), Democratizing technology: Pleasure, utility and expressiveness in DIY and maker practice, in 'Proceedings of the SIGCHI Conference on Human Factors in Computing Systems', CHI '13, Association for Computing Machinery, New York, NY, USA, pp. 2603–2612.
- Taylor, N., Hurley, U. and Connolly, P. (2016), Making Community: The Wider Role of Makerspaces in Public Life, in 'Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems', CHI '16, Association for Computing Machinery, New York, NY, USA, pp. 1415–1425.
- Wakkary, R., Schilling, M. L., Dalton, M. A., Hauser, S., Desjardins, A., Zhang, X. and Lin, H. W. (2015), Tutorial Authorship and Hybrid Designers: The Joy (and Frustration) of DIY Tutorials, in 'Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems', CHI '15, Association for Computing Machinery, New York, NY, USA, pp. 609–618.
- Wobbrock, J. O., Kane, S. K., Gajos, K. Z., Harada, S. and Froehlich, J. (2011), 'Ability-Based Design: Concept, Principles and Examples', *ACM Trans. Access. Comput.* 3(3), 9:1–9:27.