



University of Glasgow | School of
Computing Science

Honours Individual Project Dissertation

IMPROVING PRIMARY TEACHERS MOTIVATION FOR COMPUTING

Hamish Allan

Abstract

Primary school teachers in Scotland face many challenges when they attempt to teach computing, creating significant knowledge, confidence and cost barriers. These challenges limit the effective delivery of the computing curriculum.

This project attempts to determine if framing these challenges as a motivational problem could be used to tackle the initial pain-points that a teacher faces. This project made use of Expectancy Value Theory (Eccles 1983) to model teachers motivation, viewing teachers as learners by viewing teaching a new subject as a learner journey. A proof-of-concept website was created to allow teachers to measure their own motivation as modelled by Expectancy Value Theory and work through a set of resources each designed to tackle a specific motivational challenge identified.

While the effectiveness of this theory was not determined in this project, small-scale evaluation of the developed proof-of-concept website concluded that the approach has potential if it continues to be developed and expanded further.

Acknowledgements

With thanks to my supervisor Professor Quintin Cutts for all of his support and guidance over the course of this project.

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: Hamish Allan Date: 18 October 2025

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Project Summary	1
1.3	Report Structure	2
2	Background	3
2.1	Computing Curricula and Frameworks	3
2.1.1	Broad General Education (Scotland)	3
2.1.2	Informatics Reference Framework for Schools (Europe)	4
2.1.3	K-12 Computing Science Framework (USA)	5
2.2	Why is Computing Needed	5
2.2.1	Core Skills	5
2.2.2	Computing and AI	6
2.3	Summary	8
3	Problem Analysis	9
3.1	Pain Points	9
3.1.1	Unclear Computing Curriculum	9
3.1.2	Limited Time for Computing	10
3.1.3	Lack of Access to Computing Resources	10
3.1.4	Lack of Teacher Understanding or Confidence	10
3.1.5	Lack of Understanding of the Importance of Computing	11
3.1.6	Lack of Support	11
3.2	Teachers as a Common Factor	11
4	Solutions in Literature	13
4.1	Professional Development	13
4.1.1	In-Person PD - Developing Programming and Computational Thinking Skills	13
4.1.2	Online PD - Developing Scratch Skills	14
4.1.3	In-Practice PD - Integrating Computing and Engineering	14
4.1.4	In-Person Professional Development Hubs	14
4.2	Summary	15
5	Solution Analysis	16
5.1	Expectancy Value Theory	16
5.1.1	Perceived Value of Computing	17
5.1.2	Perceived Expectation of Success with Computing	18

5.2	Design Implications	18
6	Solution Design	20
6.1	Teacher's Motivation	20
6.1.1	Improving Value	20
6.1.2	Improving Expectation	21
6.2	Solution Requirements	21
6.3	Solution Design	22
6.3.1	Site Design and Flow	22
6.3.2	User Interface Design	23
6.3.3	Resource Design	24
6.4	Design Summary	25
7	Implementation	26
7.1	Implementation Details	26
7.2	Development Process	26
7.2.1	Sprint 1	26
7.2.2	Sprint 2	27
7.2.3	Sprint 3	27
7.2.4	Sprint 4	28
8	Evaluation	30
8.1	Methodology	30
8.2	Results	30
8.2.1	Participants	30
8.2.2	Underlying Theory	31
8.2.3	Resources	31
8.2.4	Website Design	33
8.3	Discussion	33
9	Conclusion	35
9.1	Summary	35
9.2	Reflection	35
9.3	Future work	36
	Appendices	37
A	Expectancy Value Cost Survey Questions	37
B	Website Wireframes	38
C	Video Resource Set	41
D	Slide-Style Resource Set	42
E	Evaluation Resources	43
E.1	Introductory Script	43
E.2	Interview Format	44
E.3	Debriefing Script	45

Bibliography

1 | Introduction

Computing as a discipline has had a major impact on the world, that is continuing to grow with new innovations. This increasing importance of the discipline highlights the importance of teaching the fundamental concepts, skills and processes of computing at all levels of compulsory education.

Many countries have introduced and developed computing curricula, however often they have faced challenges to properly implement them (Rich et al. 2019). The situation in Scotland is no different. While the Scottish curriculum is designed to introduce students to the basics of problem solving for computing during the Broad General Education (BGE), from ages 3 to 15, in practice this is often not the case, with inconsistent progression and limited focus on the full process (STACS 2025).

1.1 Motivation

Personal experience of computing education in Scotland at both primary and secondary school highlighted a number of problems with the subject from the perspective of a student. Often students would be taught the same content over and over again during primary school, making use of the same set of scratch tutorials and then covering all of that content again in secondary school. This is supported by a number of papers that report that teachers often struggle to design or find suitable resources for their students Greifenstein et al. (2021); Sentance and Csizmadia (2017).

Further during primary school there was no sense of progression or development of computing knowledge or skills, and no consistency of what computing was taught, leading to repetition of material in secondary. Alongside this lack of progression, computing was rarely used as a creative subject during primary or secondary education. This made a subject that is fundamental about developing creative solutions appear boring and restrictive. These observations suggest that there are a number of delivery and structural issues surrounding the Scottish computing curriculum, especially at primary school level, where teachers are not subject specialists.

1.2 Project Summary

It is clear that there are issues with the delivery of the Scottish curriculum that need to be addressed. The purpose of this project is to analyse the current approach to computing at primary school in Scotland and the issues faced by teachers when delivering computing. By reviewing and analysing computing education literature a set of pain points are developed that represent the challenges that primary school teachers may face. From these pain points teachers and their experiences and values were identified as a common factor, that should be tackled in order to improve computing in primary. Tackling teachers perception of these pain points was viewed as the initial step towards tackling the pain points, framing the problem as a motivational challenge and teachers as learners at the start of a learner journey towards teaching computing. This allowed Expectancy Value Theory to be used to model both the motivational challenges and the initial

solutions to them. A proof of concept resource is then developed to enable teachers to model and tackle the motivational barriers that they face, evaluated by stakeholders from computing education improvement.

1.3 Report Structure

The next chapters of this paper will develop the underlying problems that teachers face introducing how framing them as a motivational learner challenge can theoretically improve their experience and develop a proof of concept to demonstrate this theory.

Chapter 2 : introduces background information explaining how different computing curricula are structured and why it is important to teach and improve computing in schools.

Chapter 3 : presents an analysis of the pain points that can be identified from the literature and how teachers are a common factor.

Chapter 4 : discusses how existing solutions attempt to tackle these pain points and the limitations with current approaches.

Chapter 5 : frames the initial solution as a motivational challenge and discusses how expectancy value theory works to model each of the identified pain points.

Chapter 6 : creates a design outline a concrete implementation of this theory and defines the core requirements and structure of such implementation.

Chapter 7 : presents a specific proof of concept implementation of the design discussing how the implementation evolved during the development process.

Chapter 8 : presents a small scale evaluation of the proof of concept and the underlying theory to develop an understanding of the future direction of the theory and concrete implementations

Chapter 9: summarises everything that was achieved during this project, presents a reflection of the overall project and outlines what future work that can be done.

2 | Background

This chapter provides an overview of three current computing curricula, how they are structured and their level of detail alongside a justification on why computing is an important subject looking at its skill building potential and the continued need for the subject with the development in artificial intelligence.

2.1 Computing Curricula and Frameworks

Education is usually defined at a national level, with different Governments defining their own systems, which in turn means each has their own computing education curriculum. This paper mainly refers to the Scottish Broad General Education [BGE] system¹ however other computing education frameworks are now explored in order to develop an understanding of the similarities and differences in approach, content and depth. The two additional frameworks reviewed are the EU's Informatics Reference Framework for School [IRFS], and the USA's K-12 Computing Science Framework [K-12 CS].

2.1.1 Broad General Education (Scotland)

BGE computing is taught from nursery level (aged 3) until the third year of secondary school (aged 14), however this dissertation will focus only the primary school levels; Early, First, and Second Level², representing ages 3-5, 5-9 and 9-13.

The goal of the computing curriculum is to develop students' understanding of the complete computing process, and develop related skills. Students should be able to analyse and extract relevant information from a problem, then identify relevant tools and methods to use to develop a computing solution, and finally validate their solution against the initial problem (Connor et al. 2017). This is conveyed by splitting computing into three organisers:

Understanding the World Through Computational Thinking Being able to look at the world around you and identify processes and information that exist. Being able to extract relevant patterns, characteristics, structure and steps from processes and information.

Understanding and Analysing Computing Technology How information and processes are represented by humans and computers. What tools and mechanisms can be used to represent different processes and information. Being able to read and understand program sections and predict output.

Designing, Building and Testing Computing Solutions Putting together the information and processes, and tools and mechanisms to create computing solutions. Being able to represent and explain problems using computing terms and concepts. Using computing tools to represent information. Being able to use and translate different representations of information and processes.

Connor et al. (2017); Education Scotland (2017)

¹<https://education.gov.scot/curriculum-for-excellence/about-curriculum-for-excellence/curriculum-stages/broad-general-education/>

²<https://education.gov.scot/parentzone/curriculum-in-scotland/curriculum-levels/>

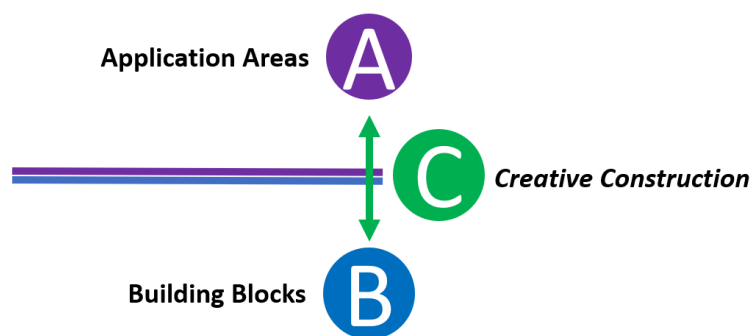


Figure 2.1: The ABC Model. A framework describing the different stages of the modelling process, and the relationship between them (Cutts 2024).

Each organiser is further split into expectation and outcome statements, with 1-3 statements per organiser per BGE level.

The ABC Model The BGE curriculum and organisers are designed using an underlying modelling framework, used to represent the different parts of the modelling process that exists in computing (and other subjects). This framework is called the ABC Model (Connor et al. 2017; Cutts 2024).

The model (Figure 2.1) has 3 sections each describing a different part of the modelling process. It is vital to understand all of the different parts and the relationships between them in order to successfully model a solution for a given problem. The stages of the model are as follows:

- [A] **Application Areas:** The context the task/problem exists within, the context of the thing to be modelled. The context being modelled needs to be explored to understand what features could be modelled.
- [B] **Building Blocks:** Systems of materials, tools and methods that can be used to create/solve/-model the task. In order to be able to identify these features the individual needs to understand what materials, tools and methods exist, and what they can model.
- [C] **Creative Construction:** This is the modelling process, where we take the task in a set context or domain (A) and model it using the materials, tools and methods that we have for a given system (B).

A key step in modelling is the creation of a connection between the feature you want to model in A, to relevant uses of the modelling tools, materials and methods in B. Once these connections between the features and tools have been made, the model can be built.

The final step of the process, is evaluating and reviewing the created model, to see if it is a good model of the original thing. What a good model looks like is specific to the context or domain of the thing (A).

The publicly available definition of the curriculum is only 2 pages long and contains only high level learner statements, leaving the concrete implementation up to the teacher to interpret relying on their understanding of the ABC model to implement the curriculum as intended.

2.1.2 Informatics Reference Framework for Schools (Europe)

The EU's IRFS aim is to provide a high level framework for EU member states to follow when implementing informatics education into their own education systems.

The framework aims to unify Europe's informatics education, under a shared set of aims, objectives and topic area. The framework aims to develop students informatics concepts, alongside

underpinning knowledge and skills. The IRFS defined a set of aims and objectives that students should have met by the end of compulsory education.

- Use digital tools in a conscious, responsible, confident, competent and creative way.
- Understand the phenomena, concepts, principles and practices of informatics and the multifaceted ways of applying them to model, interpret, and operate on reality.
- Analyse, design, frame and solve problems by devising representations, designing algorithmic solutions and implementing these in a programming language.
- Develop computational models to creatively investigate, understand and communicate about natural and artificial phenomena and systems.
- Identify, analyse and discuss ethical and social issues associated with computational systems and their use as well as their potential benefits and risks.

Caspersen et al. (2022)

Due to the IRFS being designed for governments to develop into their own curricula and not for teachers to directly design lessons on, the framework is only defined at a high level and contains very limited detail on what it would look like during concrete lessons, to allow it to fit into the different education environments that each EU state has.

2.1.3 K-12 Computing Science Framework (USA)

The USA's K-12 CS Framework is a complex computing framework designed for states to adopt individually. to give a consistent computing education across the USA. It aims to enable students to effectively use CS in their education and future workplaces and careers (CSTA et al. 2016).

The curriculum is split into 5 core concepts, 5 cross-cutting concepts, and 7 core practices. The framework defined a set of aims and objectives that students should have met by the end of compulsory education.

- Critically engage in public discussion on computer science topics.
- Develop as learners, users, and creators of computer science knowledge and artifacts.
- Better understand the role of computing in the world around them.
- Learn, perform, and express themselves in other subjects and interests.

CSTA et al. (2016)

The K-12 CS framework specification document is over 300 pages long, containing very specific technical information and design details. While this can be helpful for a teacher that has a strong understanding of computing, it creates a number of barriers for a new teacher as they try to read and understand the full curriculum.

2.2 Why is Computing Needed

It is important to illustrate why computing is an important subject before exploring how computing can be improved. It is also important to justify why computing should be taught at all levels of education, even for people who will never take it further. This section also identifies how current computing curricular attempt to develop these important factors.

2.2.1 Core Skills

The Scottish Curriculum, as discussed in subsection 2.1.1 is designed to teach students a well rounded approach to tackling Computing problems, which in turn develops a number of computing skills and concepts (Connor et al. 2017). One of the core goals of the curriculum is developing a students ability to understand problem solving processes by developing the students

understanding of how problem solving works, developing their computational thinking skills and basic problem solving abilities.

Computational Thinking [CT] is a way of thinking, and associated set of core skills, defined by Wing (2006). These skills give a person an understanding of how to understand and represent all of the processes and information happening in a specific context. While many of the skills CT develops have significant importance within computing, they are not exclusive to the subject, with many of these skills used in everyday life, to understand and process the world around us (Wing 2006; Salehi et al. 2020; Arfé et al. 2020; Flórez et al. 2017). Computational Thinking is a core aspect of the Scottish curriculum, in particular through the first organiser "Understanding the World through Computational Thinking". CT skills can also be developed and embedded in many different subject areas, which in turn enables the enhancement of a student's understanding of that subject (Marcelino et al. 2018; Bocconi et al. 2022; Flórez et al. 2017). These skills are not just useful in education, but also beyond in the workplace, with a basic level of computing and CT knowledge providing benefits and increasing "executive functions" for all industries not just computing (Vegas et al. 2021).

The development of problem solving skills is another key element of computing education, and another core aspect of the Scottish computing curriculum. In Scotland the focus is on developing students' skills and knowledge in the full problem solving process, as opposed to only looking at solution generation. The full process involves looking at, and extracting important details about the information and processes that a real-world problem has, then identifying relevant tools and processes to develop a domain (computing) focused solution to that problem, and then evaluating the effectiveness of that solution (Connor et al. 2017; Arfé et al. 2020). Some studies have shown that Computing Science may be more efficient at developing problem solving than other STEM subjects (Vegas et al. 2021; Salehi et al. 2020).

Developing core skills that are useful outside of computing is not unique to the Scottish curriculum, but common to most computing education curricula or frameworks. An example of this is the US K-12 Computing Science framework (as discussed in subsection 2.1.3) that shares a number of skills and practices with other K-12 STEM subjects, with the CS K-12 framework directly referencing the relationships between Computing, Maths, Science and Engineering practices, represented in Figure 2.2. This shows the overlap between the practices of these subjects, and demonstrates the cross-curricular benefits of students developing these core skills. Further Computational Thinking is referenced as an important industry skill in the US Next Generation Science Standard report, which lists Mathematics and Computational Thinking as a core industrial practice. Included as part of this core practice is Algorithmic Thinking, the ability to process and understand data, the ability to search and identify relevant information, all skills that can be developed by computing (NAP 2013).

2.2.2 Computing and AI

Recent developments in AI technology, particularly Generative AI chatbots like ChatGPT or Claude have changed how people interact with technology and raised questions on the requirement for computing as a subject (Alkaabi et al. 2025).

It is important to consider if computing education is still required, with the continuing enhancements of Generative AI, specifically considering enhancements in code generation capabilities. At the current level, AI still requires a basic level of subject knowledge to allow for a person to appropriately use AI, both when instructing the AI (prompting) and critically analysing the AI's output (Becker et al. 2023; Böckeler 2025; Gong et al. 2025; Suh et al. 2025). Research has suggested that it is unlikely that AI will be able to undertake fully autonomous tasks in the near future, with AI needing a human to understand and explain the specific problem, and then integrate the AI generated code into existing code-bases (Böckeler 2025). This suggests that while computing may move away from a primary focus on writing programs, and more towards core

Figure 5.2: Relationships between computer science, science and engineering, and math practices

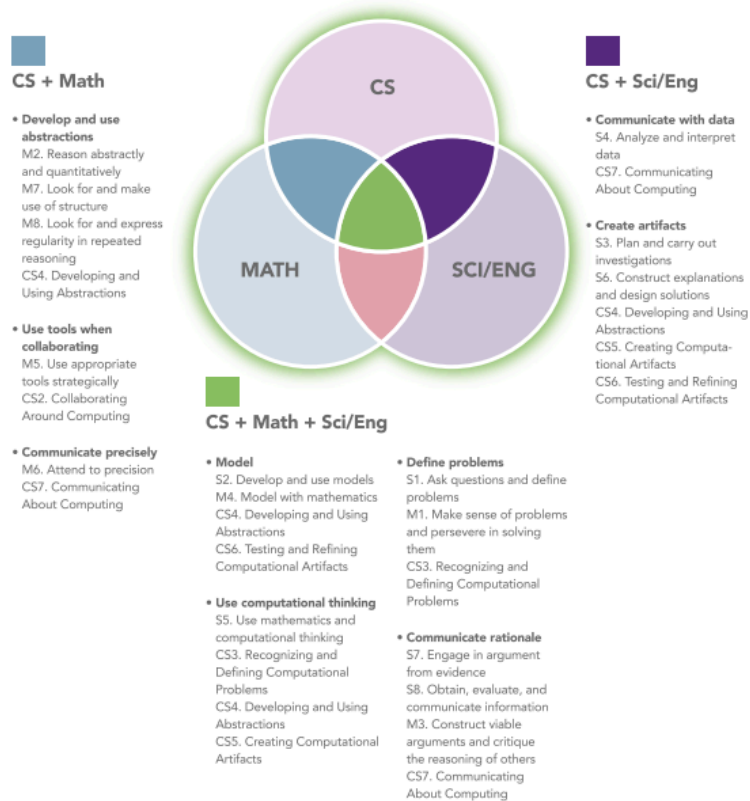


Figure 2.2: US K-12 Framework: Relationship between computing, maths, science and engineering practices CSTA et al. (2016)

computing concepts and skills to allow appropriate interactions with AI, it is unlikely computing will become completely obsolete (Böckeler 2025; Becker et al. 2023).

One of the fundamental limitations with Generative AI is the ability for its output to contain mistakes (Banerjee et al. 2025), which means that students need to have significant background knowledge in the subject, to be able to critically analyse the given output, to identify any mistakes, bias or misinterpretation of the prompt (Gong et al. 2025; Becker et al. 2023). Research on programming outputs from older AI models reported approximately 10% of python code had basic syntax error (Böckeler 2025), while newer models have likely improved this, it shows how important it is to evaluate an AI's output.

Another factor supporting the requirement for computing in a world with AI, is the importance of ensuring that everyone understand how to use AI in a safe, responsible and efficient manner (DfE 2025). Due to the ease of use and prevalence of AI chatbots, it is likely that most people will use or interact with an AI at some point (Alkaabi et al. 2025; World 2025; EuroStat 2025), suggesting that computing education should teach students to use AI safely and responsibly. The importance of AI safety is echoed by many national governments who identified this as a top issue, especially around data protection, privacy, accuracy and transparency (OECD 2023).

It is important that during computing education, when developing students understanding of how computers interact and process data, that students are also taught how an AI use and process their input, often being stored for use in analytics or training data (OECD 2023). By teaching this process, students should understand why they should not share personal or sensitive information with an AI (DfE 2025). Students should also be aware of inherent bias that GenAI might have, and how they can spot this in its outputs (OECD 2023).

Many of these points are not specific to computing, but apply generally to other disciplines as well. This further highlights the importance of computing education, both to create an understanding of problem solving and to develop safe and effective AI usage, using a core understanding of how various computing technologies work.

The above points have all focused on the current state of AI, but AI is a fast developing field and generative AI keeps getting more powerful, while current research suggests that AI will not replace the need for computing, it is important that the subject adapts to future changes.

2.3 Summary

While each of the three curricula analysed all have different fundamental designs they have similar focuses on what skills and concepts should be covered. The main difference is that the Scottish curriculum is organised around the development of students understanding of the ABC modelling process, opposed to the other curricula organising around core computing topics.

Alongside the curricula the importance of computing was investigated, with the importance being demonstrated by the range of different core skills that computing develops which are useful not just for computing but for a wider range of subject areas.

Recent develops of AI technology do not remove the need for computing but further emphasises the importance of the subject. In order to use AI effectively and safely students require a significant understanding of core computing concepts.

3 | Problem Analysis

There is a lack of research in computing education that identifies and consolidates all of the issues that primary school teachers face around computing science.

3.1 Pain Points

Through analysis of the literature in computing education, specifically looking at professional development and teacher's experience in computing, some common problems can be identified and analysed. While these pain points are experienced by teachers in many different countries, for this project they have been analysed under the specific context of the Scottish BGE Curriculum.

- Unclear Computing Curriculum
- Limited Time for Computing Learning
- Lack of Access to Computing Resources
- Lack of Teacher Understanding or Confidence
- Lack of Understand of the Importance of Computing
- Lack of Support when Teaching Computing

3.1.1 Unclear Computing Curriculum

Many computing curricula suffer from a lack of concrete information for teachers to follow, and an overuse of technical and subject specific terminology teachers may not understand.

A lack of detail in the BGE curriculum specification for computing can make it challenging for teachers to interpret. The computing curriculum specification (expectations and outcomes table) contains only 1-3 high level learner statements per organisers per level, with 3 organisers in computing and each level encompassing 2-3 years of education (Education Scotland 2017). This is significantly less information than other curriculum specifications (outlined in section 2.1) and lacks any concrete information for teachers explaining what these statements look like in practice, making it difficult for the teacher to cover during lessons.

Alongside this while the BGE curriculum uses less subject specific language than other frameworks, notably the US K-12 CS Framework, it still makes use of subject specific terminology that without definition, teachers may not understand, thus making it even harder for them to cover during their lessons. An example of this is learning statement TCH 1-13a where the term "Core computational thinking concepts" is never defined.

I can explore and comment on processes in the world around me making use of **core computational thinking concepts** and can organise information in a logical way.

TCH 1-13a - Education Scotland (2017)

In addition to the curriculum itself, there are issues when teachers attempt to measure student progress in the computing curriculum. There is no standardised testing for computing in BGE, with teachers being responsible for evaluating their student's progress using their own professional

and subject knowledge (Education Scotland 2021). Due to the fact many teachers do not have much subject knowledge, their ability to evaluate their students is going to be limited. This means the teacher is unable to evaluate how well their lessons are developing their students computing ability.

3.1.2 Limited Time for Computing

The small amount of time available for computing makes it difficult to gain and build skills and conceptual understanding.

While there is no official time allocation for each of the organisers in Scotland most primary school contact time is taken up by the development of literacy and numeracy skills, as these are the core areas that the curriculum monitors. Roughly 2/3 of contact time is dedicated to literacy and numeracy content, with the other 1/3 dedicated to the other organisers, which results in around 15–20 hours a year available for computing which translates to around an hour long lesson every 2 weeks (Cutts, personal communication, 2025). In reality it is possible the amount of time spent on computing is even lower due to all of the challenges around teaching computing.

The amount of time spent on computing during BGE is not enough to allow students to develop a strong foundation in computing Rich et al. (2021), and is significantly less than the 1–3 hours that teachers across Europe (including Scotland) believe should be dedicated to computing (Nardelli et al. 2025). In addition the small amount of time that is spent on computing can make teachers believe that computing is not a major subject area, and does not have much importance for their students (Society 2017; Tsui 2023), which is not the case as outlined in section 2.2.

3.1.3 Lack of Access to Computing Resources

Teachers often do not have access to or knowledge of high quality resources to support in the delivery of computing.

A common issue reported in literature is teachers' lack of knowledge or access to high quality computing resources for their relevant curriculum (Rich et al. 2019; 2021; Ramirez-Salgado et al. 2025). Because teachers often do not have the knowledge to create their own resources, they may have to seek out resources from internet resource hubs which can add an additional challenge, especially if the teacher is unaware of where to look. While there are many resources available on the internet when these resources come from a source the teacher does not know, or if they are not designed for the BGE curriculum, the teacher has to evaluate each resource for quality and relevance (Sentance and Csizmadia 2017). This process is both time consuming for the teacher and difficult as it requires an understanding of computing concepts and what the BGE curriculum means, which many teachers do not have.

3.1.4 Lack of Teacher Understanding or Confidence

Teachers do not have an adequate understanding of computing, and therefore struggle to teach it effectively.

Studies have shown that many teachers feel they do not have adequate knowledge to teach computing effectively, with them struggling to understand the computing concepts they are meant to be teaching (Greifenstein et al. 2021; Rich et al. 2019).

Further to this teachers report a need to ensure that they are familiar with the content being taught, including identifying common pitfalls that their students may experience. In order for teachers to be able to effectively support their students during a lesson they also want to learn material outside of the lesson's scope, attempting to keep their own knowledge ahead of and broader than their students (Rich et al. 2021). This is challenging when the teacher does not fully

understand the content, and they will have to seek out support, development opportunities or teach themselves, which require a commitment that a teacher may feel is unfeasible.

Aside from the knowledge barriers there is also a confidence challenge related to teacher's knowledge. If a teacher does not feel they have an adequate knowledge of computing they are unlikely to have confidence in their ability to teach computing properly (Rich et al. 2021; Greifenstein et al. 2021; Learning Directorate 2012), with a specific concern around the potential for something to go wrong, and the teacher being unable to recover from an error, or effectively help their students (Greifenstein et al. 2021). Alongside this it is common for a teacher to feel that their students already have more knowledge than them, and therefore they will struggle to teach anything additional or useful during lessons Rich et al. (2019).

These factors all create barriers to a teacher's belief that they are able to teach computing well.

3.1.5 Lack of Understanding of the Importance of Computing

While computing can be used to develop and build a number of core foundational skills this is not always known by teachers.

As discussed in subsection 2.2.1 computing develops a number of core skills, however teachers are not necessarily aware of the skills and concepts that computing education can develop, and the usefulness that these hold for their student, even outside of computing science (Larke 2019; Mason and Rich 2019). As teachers are not made aware of the benefits of computing (as discussed in section 2.2), they are unlikely to understand the importance of teaching of computing, and therefore may not feel it is important to try and overcome these barriers to the subject.

In addition computing has a number of benefits when integrated with others subjects, where it can act as a vehicle for developing subject specific knowledge in a wide range of subjects (Society 2017). Many teachers do not attempt to integrate computing with other subjects, teaching it purely as a stand-alone subject (Rich et al. 2019) suggesting that they don't understand the benefits of integrating computing and/or don't have the knowledge required to integrate the subject.

3.1.6 Lack of Support

Teachers can feel they are isolated and there is no support or help when they try and teach computing.

Many teachers feel a sense of isolation when teaching computing science, often reporting that they are one of few "computing specialists" within their school (Rich et al. 2021; Sentance et al. 2022; Cutts et al. 2017). On the other hand, teachers without computing knowledge would benefit from support when beginning to teach the subject report a lack of support from their school's leadership Sentance and Csizmadia (2017). This is supported by the fact many teachers report their colleagues are "cautions" about engaging with computing (Rich et al. 2019). Again this creates additional barriers to teachers thinking about teaching computing, creating a perception that the support they need to teach it effectively, does not exist.

3.2 Teachers as a Common Factor

Many of the pain points are either directly or indirectly related to primary school teachers and their experience, values and confidence towards teaching computing, with many overlapping themes between distinct pain-points. These pain points highlight the common issues within computing education that create barriers to teachers being able (both practically and mentally) to successfully deliver computing lessons, with specific and recurring issues around a lack of education and support.

It is important to look at these issues that primary teachers face as part of making improvements to computing education, as a computing curriculum is only as good as the teachers that are being tasked with delivering it, and who currently do not have the information, education or support to do so.

Teachers should be included and engaged with as part of this processes, and made aware of why it is important to improve computing, showing them the importance of the subject and giving them the knowledge and confidence to teach the subject.

Before attempting to develop my own solution for these issues, it was important to consider existing solutions that have been undertaken before to try and solve some or all of these identified issues.

4 | Solutions in Literature

Many challenges were identified from the literature in chapter 3, with teachers knowledge, value and confidence appearing as common factors within these challenges. Several studies in the literature investigate different methods to tackle the challenges that teachers face when teaching computing.

4.1 Professional Development

There have been a number of studies and literature exploring the benefits of professional development [PD] aimed at improving computing at primary level. A number of different methods have been explored by these studies including an in-person PD course and an online-asynchronous PD course both aimed at improving teachers programming and computational thinking skills.

4.1.1 In-Person PD - Developing Programming and Computational Thinking Skills

To determine the effectiveness of PD on teachers understanding of and self-efficacy towards programming and CT a study was conducted using a year long Continuous PD [CPD] program, in the US. The program consisted of a series of in-person sessions conducted by a third-party group developing teachers knowledge and confidence through demonstration of Scratch/ScratchJr lessons, alongside discussions on concepts, methods and technologies (Rich et al. 2021). Alongside the sessions participants were given access to set of resources for the K-12 curriculum. The study reported an increase in both teachers confidence and ability in both programming and CT, with an average increase of one point on a six point self-efficacy scale. with teachers reporting that the in-person sessions and hands-on experience were the main causes for this increase, alongside access to resources and support.

Further the study reports that a primary motivator for teachers persisting with the challenge of learning and teaching computing was because of student's own interests and achievements when working on CS (Rich et al. 2021). This suggests that before large scale upskilling/improvements can take place, teachers need to be sufficiently motivated to improve the effectiveness and uptake of such programs.

While this form of CPD program appears to tackle a number of the pain points identified by providing teachers with knowledge, experience and support for computing there are some challenges faced. The study reported that many of the participants were motivated to persist with computing and undertake development opportunities however due to the pain points identified it is unlikely that all teachers will feel this way. This suggests that improving teachers motivation could be a pre-condition to large scale CPD programs.

Secondly as this program uses in-person sessions there would be a number of logistical and cost challenges before a wide-scale rollout could happen. Central locations that allow teachers from all over Scotland would have to be decided, as well as providing compensation (payment) for transport and time costs. The logistic and cost challenges are further emphasised by Scotland's remote nature (Rich et al. 2019).

4.1.2 Online PD - Developing Scratch Skills

Looking at a different method of PD (Marcelino et al. 2018) explored a 54-hour long online asynchronous program, aimed at developing teachers understanding of Scratch and computational thinking skills. The course was designed to allow teachers to work through each section within their own time, with a set date-range for each section. The course was made up of 8 hours of core concept explanation, 6 hours of scratch exploration, 26 hours of scratch problems and then 14 hours for a final scratch project. The study reported a high level of satisfaction among the participants, and a positive impact on their scratch and CT skills. The study also reported the amount of hands on experience participants revived as a key positive of the program method.

This PD program focused purely on developing teachers programming experience and knowledge, which while important is only one part of the computing science curriculum. There are also many other concepts and skills that are taught through computing that a teacher would require training in. Alongside this the program only solved one of the identified problems, developing the teachers understanding of the subject, but not working to tackle any of the other issues.

Further while the online and asynchronous nature of the program make the style more feasible to rollout across a rural country (Rich et al. 2019), the length of the program required a significant time commitment from teachers, which as discussed by Rich et al. (2021) in subsection 4.1.1, will likely require teachers to have a high motivation towards computing before taking part.

4.1.3 In-Practice PD - Integrating Computing and Engineering

The previous two sections have focused on improving teachers knowledge, skills and experience in computing, however PD studies have been done with different focuses. Another focus point is showing teachers that computing is engaging, especially when integrated into projects with other subject areas. A study by Bartholomew et al. supported teachers through a joint engineering and computing project, with sections of art and literacy, that got students to design, build and program a basic light up digital storyboard. The study was aimed at both improving teachers engagement with computing and proving that students benefit from integrated projects (Bartholomew et al. 2023).

The study reported that students attitude towards coding increased after taking part in the digital storyboard project, suggesting that this style of integrated learning would work to boost student attitude and engagement towards the subject. On the other hand while the study demonstrated to teachers the power of integrating computing with other subjects, it did not adequately develop their understanding of either subject (computing and engineering) in order to allow them to do this in the future, with teachers who were interviewed reporting that they would not be able to independently run a project like this.

This highlights how important it is to tackle teachers lack of computing knowledge alongside tackling the other pain points identified.

4.1.4 In-Person Professional Development Hubs

Another style of PD model is an in-person HUB based model, where teachers attend sessions run by other teachers at set hubs around an area, with a focus on upskilling and building a network and community of teachers. This was done for computing in both England and Scotland in the mid 2010s (Rich et al. 2019) however due to the scope of this project only the Scottish model was analysed in more detail.

The Scottish Hub system as studied by Cutts et al. (2017) established a number of PD hubs across Scotland called PLAN C. These hubs focused on developing secondary teachers computing abilities, specifically around Scotland's newly implemented curriculum.

The program received good feedback from the teachers that took part with teachers gaining a deeper understanding of both computing knowledge, computing educational theory and methods of delivering computing lessons. The system did however struggle with attendance issues, with many Hub leaders reporting attendance as a key problem, with 50% of computing teachers attending at least one session, but only 25% attending enough to become certified. There could be a number of reasons for this low attendance, with one issue being the geographical issues that Scotland faces due to its rural nature, making centralised hubs difficult to get to for many teachers. As well these hub sessions were run in evenings and required a large time commitment from teachers, suggesting again that teachers would need to be motivated and understand the value of computing to attend these sessions.

Looking at this style for primary teachers it would likely suffer from the same scalability issues as an in-person PD program would. While this was rolled out for secondary teachers there were only around 640 of them when this survey was done, compared to the ca. 24,000 primary school teachers that there are in Scotland. As well it would be challenging to attract non-specialist primary school teachers to take part due to the large time commitment required, and teacher's lack of understanding of the importance that teaching computing, and subsequently improving their own skills holds for their students.

4.2 Summary

While many studies have been done which look at improving computing in schools, they often focus on only on a small subset of the pain points, and do not look at the full problem. These studies could have been more holistic and attempted to tackle the common factors causing these pain points, because of the interconnected nature of the pain points.

Further many of these studies would suffer from scalability issues if their models were rolled out across Scotland, making them unfeasible solutions. On top of this many of these methods would require a significant time commitment from the teachers taking part, which presents a logistical and motivational barrier to engagement. It can then be reasoned that alongside offering training and resources it is important to consider how a teacher's understanding of and motivation towards computing can affect their approach to improving their delivery of the subject.

This is backed up by Rich et al. (2021) finding that teachers participating in their professional development program were motivated by their students enjoyment and attainment in computing. This shows the importance of motivation and understanding of the benefits of computing and is supported by other papers showing how important it is for teachers to be motivated and understand the importance of computing in order to work with them to improve the subject through professional development programs or otherwise (Rich et al. 2019; Ramirez-Salgado et al. 2025).

5 | Solution Analysis

Many current solutions suffer from similar limitations, mainly that they only focus on a small part of the wider problem and do not attempt to tackle all of the identified problems, their root causes or underlying themes. Many existing solutions also suffer from scalability and engagement issues which makes them infeasible to implement across Scotland.

As discovered in chapter 3 teachers are central to improving computing education in primary school, and it is important that they understand how to teach computing, why it is important to teach computing and that it is possible to teach computing. The initial step to solving many of these problems is a motivational challenge. This identification as a motivational challenge is supported by PD literature reporting that motivation is a significant factor in getting teachers to engage with computing (Rich et al. 2021).

Having identified the problem as a motivational challenge, a motivational framework/modelling method was needed to start developing a deeper understanding of the problem and begin to develop a solution. Through previous work I was aware of Eccles's Expectancy Value Theory [EVT], which fits well into the problems teachers had around the importance (value) and confidence (expectancy of success and cost) when teaching computing. This theory was designed for modelling students motivation in a learning context or journey, so does not initially fit the context here, however there is a learner journey associated with teaching any subject, especially when attempting to teach a subject they do not have a background in. A teacher who is trying to teach computing for the first time, can be considered to be at the start of a significant learning journey.

Before settling on EVT other motivational theories were considered. Tinto (1975) model of student retention was considered to model how the lack of support and isolation that teachers experience affect how they approach computing. This model was not chosen for this project as it does not model the correct part of the teachers learning journey. The retention model focuses on people when they are progressing through a learning journey, in this case once a teacher has started teaching computing however many of the identified issues effect teachers before or as they are starting their learning journey. Retention based models could be used as a follow through to this project, but would not work as well at this initial stage.

Another motivational model considered was Elliot and McGregor (2001) 2×2 Achievement Goal Framework, based on wider Achievement Goal Theory. This defined a motivation matrix defining a persons motivation towards a goal based on a goals performance destination and effect. While this theory can be used to mode the initial phase of a learning journey, it focuses mainly on the teachers ability and importance towards computing, both of which are considered by EVT, alongside additional motivational factors. Therefore it was decided that EVT would model a broader range of factors than Achievement Goal Theory.

5.1 Expectancy Value Theory

Expectancy Value Theory [EVT] break a person's motivation down into two primary factors, expectation (of success) and value (of learning). These factors effect a person's achievement, effort and persistence for a given task (Eccles 1983; Wigfield and Eccles 2000).

The theory breaks down value into 4 sub-values, which are defined for this specific use-case below

Attainment Value The importance of doing well in a given task.

Intrinsic Value The enjoyment obtained from performing a given task.

Utility Value How well a task fits into future plans and goals. The usefulness.

Cost How one task may limit engagement in others, how much effort is required, what is the emotional cost.

Eccles (1983)

The pain points identified in chapter 3 can be categorised into EVT, allowing the modelling of teachers motivation towards computing.

5.1.1 Perceived Value of Computing

Attainment: Attainment value is the level of importance that a teacher feels to do well in the delivery of the computing curriculum, and how measurable that success factor is for the teacher to understand and gauge their success, at both a personal and institutional (eg their School) level.

Teacher's attainment value is likely to be effected by the limited time dedicated to computing and the lack of support and isolation that teachers experience when planning and delivering computing lessons.

As computing is not a major part of the Scottish curriculum teachers may not perceive limited negative consequences to their students if they do not fully teach the subject. This is reinforced by the lack of centralised monitoring or testing of computing, meaning teachers are unable to effectively evaluate their students attainment in the subject.

Further as teachers often feel isolated when teaching computing, and there are limited number of teachers engaging with the subject there is not much institutional importance for the teacher to upskill themselves and ensure they are teaching to the same standard as the rest of the teachers in their school.

Intrinsic: Intrinsic value is the enjoyment obtained from performing a given task. In this context the teachers enjoyment when creating and delivering computing lessons.

Teachers intrinsic value can be effected by their lack of confidence around teaching the subject, with teachers struggling to develop a level of knowledge that they feel is adequate to both teach and support their students. This lack of confidence a teacher feels around their abilities is likely to reduce the enjoyment they feel around both the planning and delivery of computing lessons.

Utility: Utility value represents how useful the content and skills taught though computing education are thought to be for a teachers students, both in their current education and beyond. It also represents the teacher's perception of their students' enjoyment of computing lessons.

Teachers often do not understand the range of skills and knowledge developed by computing and as such they are unlikely to believe that computing holds much value for their students, especially outwith computing science. This is further emphasised by teachers not attempting to integrate computing with other subjects demonstrating the lack of perceived usefulness of this practice.

As well even if a teacher holds a high utility value towards computing because of their lack of subject knowledge and perception that they won't be able to teach the content properly, they are unlikely to think they will be able to teach their students the content in a correct or useful manner.

Cost: Turning to the final value factor described in Expectancy Value Theory, the cost of preparing for and teaching computing lessons, both the expected cost, what the teacher perceives as the cost ahead of time, and the real cost of the task. While these values can be different, it is important to attempt to lower both of them (in sync), to make computing a more appealing subject to teach, and therefore increasing motivation.

The main cost associated with computing is the resource and material preparation required for a computing lessons, which takes a significant amount of time and effort, which teachers often do not have. This cost is higher if the teacher does not know where to look for resources or does not have enough subject knowledge to create or evaluate resources.

Teachers also need to spent a significant amount of time learning computing to a level they feel is adequate to be able to effectively teach their students, often requiring them to learn the content from scratch.

5.1.2 Perceived Expectation of Success with Computing

Expectation of success is the second core factor in Expectancy-Value Theory, and an important aspect of teachers motivation, specifically focused on present factors (Eccles 1983; Wigfield and Eccles 2000). Teachers have reported numerous challenges and struggles related to developing and delivering CS lessons, which all create a general low expectation that they will deliver a successful computing lesson.

One factor influencing low expectations is teachers lack of subject knowledge and confidence in their ability to teach computing, with teachers often not feeling that they have adequate knowledge to deliver successfully and useful lessons for their students.

Another factor effecting expectations is teachers lack of confidence in their ability to support students, especially when coming across bugs, technical issues or unexpected behaviour. This then leads to concern that students experiencing serious issues could derail the teachers lesson.

Finally teachers do not have access to or knowledge of high quality teaching and support resources for the relevant curriculum so they have to rely on either creating their own resources based on limited curriculum guidance or adapting content found though internet searches, which may not be from the correct curriculum or level. Due to the limited knowledge that teachers posses for CS they will likely struggle to correctly develop/adapt and understand the material for their lesson. If teachers are not confident in their material then they are unlikely to be confident teaching it.

All of these factors echo the core issue that teachers do not have confidence that their lessons will be successful for both themselves delivering them, and for their students learning from them.

5.2 Design Implications

By modelling the identified pain points against EVT it is clearer that these can create a motivational issue that leads to teachers having a low perceived value and expectation of success and a high perceived cost when considering teaching computing.

Any solutions must attempt to tackle these factors, increasing teachers perceived value of computing by showing the importance of teaching computing to their students, increasing their expectation of success by showing how computing can be taught successfully with limited resources and lowering cost by providing methods and resources to successfully design and teach computing lessons. The solution can be modelled similarly to the problem using expectancy value theory again, to understand how to target specific issues and improve teachers motivation.

Once teachers motivation to engage with computing has been addressed, and teachers get started on their learner journey further work can then be undertaken to ensure teachers are supported

and engaged throughout their learner journey, until they are confident teaching computing at all levels.

6 | Solution Design

As discovered in chapter 5 improving teacher's motivation towards computing is an important initial step in improving computing education in primary school.

A solution needs to be designed to target teachers motivation and work to improve it. Teachers are not the only stakeholders involved in computing education however they are the primary stakeholder for this project. While they may not be primary stakeholders it is still important to identify other stakeholders that are involved in education, defined below:

- **Non-specialist Primary Teachers** - Plan and Delivery Computing Lessons
- Support Staff - Support teachers with lessons and specific pupils
- School Leadership (e.g. Head Teacher) - Define schools curriculum priorities and support available
- Council Education Department - Provide resources and higher level priorities
- Education Scotland - Define the curriculum and key milestones for education
- Computing Resources Hubs (e.g. STACS) - Work with teachers to develop support and resources

6.1 Teacher's Motivation

Based on the analysis of teacher's motivation described in chapter 5, the different pain points and related motivational challenges can be outlined and grouped together, based on related solutions. These groups form the basis of the resource set to be developed for the solution website.

6.1.1 Improving Value

1: Communicate the core skills developed through computing

This addresses the following value challenges:

- The computing curriculum is a small area, with a limited scope and importance
- Teachers may not be aware of the skills that computing develops for their students
- Teachers may not be aware of the benefits of integrating computing with other subjects

2: Show that BGE computing can be taught without needing extensive computing knowledge

This addresses the following value challenges:

- Teachers are cautious about computing
- Computing requires a large amount of prep time, taking away from other subjects

3: Highlight the different methods and tools for teaching computing

This allows lessons to be designed to fit teachers own confidence level and computing experience.

This addresses the following value challenges:

- Teachers are cautious about computing
- Students might find the content boring, and the teacher is unable to make it engaging
- Computing requires a large amount of prep time, taking away from other subjects

4: Show that and resources are available for the Scottish curriculum and how they work

This addresses the following value challenges:

- Teachers are cautious about computing
- Teachers feel isolated and unsupported when teaching computing
- Computing requires a large amount of prep time, taking away from other subjects
- Lack of awareness of computing resources for the Scottish curriculum

5: Computing can be used as a vehicle for teaching other subject areas

This addresses the following value challenges:

- The computing curriculum is a small area, with a limited scope and importance
- Students might find the content boring, and the teacher is unable to make it engaging
- Teachers may not be aware of the skills that computing develops for their students
- Teacher may not be aware of the benefits of integrating computing with other subjects

6.1.2 Improving Expectation

1: Show that BGE computing can be taught without needing extensive computing knowledge

This addresses the following expectancy challenges:

- Lack of Subject Knowledge
- Lack of support and debugging skills

2: Show that and resources are available for the Scottish curriculum and how they work

This addresses the following expectancy challenges:

- Lack of Subject Knowledge
- Lack of awareness of computing resources for the Scottish curriculum

3: Highlight the different methods and tools for teaching computing

This addresses the following expectancy challenges:

- Lack of subject knowledge
- Lack of support and debugging skills

6.2 Solution Requirements

Based on analysis of PD designed to cater to teachers and discussions with my academic supervisor around teachers as stakeholders the following set of core requirements were produced.

Users have limited time - System should be easy to use and drop-in on

- The website should have no login
- The website should contain a series of **short** video resources
- The website should make it easy for a teacher to return to at a later time
- The website should not take more than a few hours to complete (average)

Users will have limited/no computing background - Resources should be accessible with limited computing background

- The website should be focused primarily with increasing motivation
- Resources on the website should use standard or clearly defined terminology

Users will have limited motivation for computing - Need to make the site engaging to in order to keep them using it

- The website will contain links to the literature and research, to allow users to explore further if they wish, but not overloading them with information on the main flow
- The website should create a user specific pathway through the resources (personalisation)
- The website should be easy to use and drop-in on (Requirement 1)

Users will be teaching in Scotland – The site should be specific to the Scottish Curriculum

- The website should use language and terminology relevant to the Expectations and Outcomes
- The website should refer to resources and methods relevant to the Scottish BGE Curriculum

6.3 Solution Design

Based on the requirements defined in section 6.2 the website should be simple to use for the user, with a focus on creating a simple path for the user to work through the site alongside a pleasing but simplistic user interface [UI].

6.3.1 Site Design and Flow

The site was designed to have a simple primary user flow, taking the users through the resources on the site. The site also contains an optional secondary path, which allows the user to explore the underlying theory and literature more, however this is not required to develop the users motivation.

The initial user path contained 2 main sections for the user, one targeting the value of computing and one targeting the users expectation and perceived cost of computing. The user would complete one section and then move onto the next, eventually the user would be directed to external resources to continue developing their computing teaching, with improved motivation, visualised in Figure 6.1

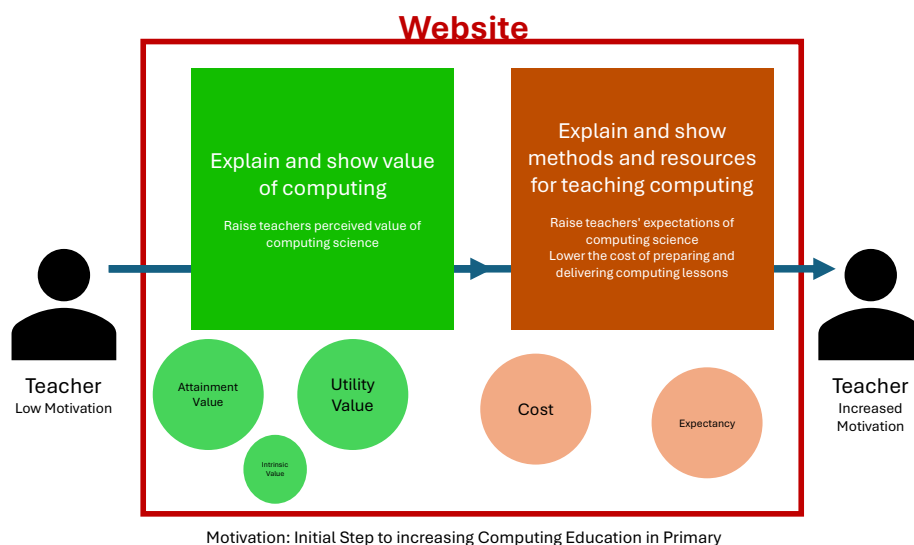


Figure 6.1: A diagram of how a user will move through the main resources of the site, first targeting their perceived value of computing and then targeting their expectation and cost.

Initial evaluation of the site design lead to a revision of the main path of the site, adding in an additional stage to create a personalised journey thorough the site. The user would complete a simple survey to gauge their motivation towards computing, modelled using EVT, which would be used to direct them to work through the appropriate resources for their modelled motivation, visualised by Figure 6.2

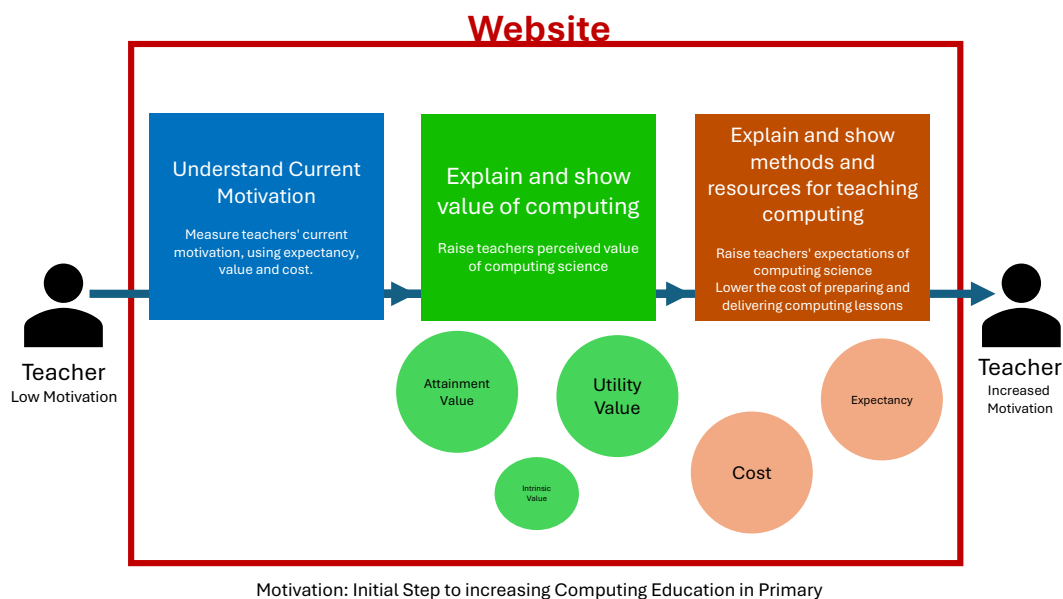


Figure 6.2: A diagram of how a user will move through the main resources of the site, first modelling their current motivation using expectancy value theory, then targeting their perceived value of computing and finally targeting their expectation and cost.

These website stages were developed into a basic site-map, with the three stages outlined in Figure 6.2 forming the pages within the main user path. The About page contains additional information about the underlying theory with links to literature. The About page is linked to from pages on the primary user path, this allows users to discover additional information about the specific part of the site they are currently at. The home and privacy pages contain additional information for the user, but are not part of the main function of the site, the sitemap is represented in Figure 6.3.

6.3.2 User Interface Design

In order to keep the site simple and intuitive to use for users when interacting with it, the UI was kept simple using standard layouts and components.

Resource Pages The main UI for the site was the different resource pages, which contain the set of resources for the specific area of EVT (as shown in Figure 6.3). The main design considerations was how each resource would be laid out, and what supporting information would be displayed for that resource.

This UI went through a major revision during the project, to reflect feedback received. The initial design used video resources, with the video being displayed on the left alongside a title, description and tags (relating to each EVT subsection being targeted) displayed on the right as shown in Figure B.1a. Multiple video resources would be displayed on a page. The user would be able to pick and choose what videos to watch and in whatever order they wanted.

A major revision of the resources on the site was agreed upon late in the project based on feedback around the video format (discussed in subsection 7.2.4). This revision led to a change in the design of the resource page, to accommodate changing the resource format from Videos to Slide Style. Alongside this a reference button was added to allow users to view the specific resources and papers used in each resource. This is shown in Figure B.1b

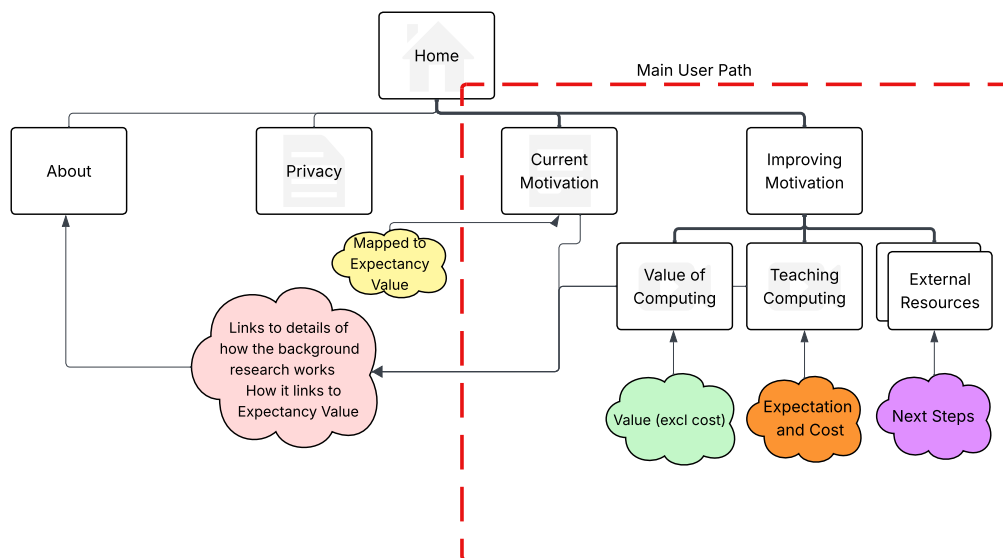


Figure 6.3: A sitemap showing the basic website layout for the solution. This highlights the main user flow and what each section is targeting within the underlying theory of the site.

Measuring Motivation The other main UI for the site was the measurement of motivation and corresponding motivational report for the user, modelled using EVT.

The first part of this process is the motivational survey, used to gather the data required to model motivation. The survey design is based on Barron et al. Expectancy Value Cost Survey of Student Motivation, which consists of 10 questions, with each question represented as a labelled scale between 1 and 6. Each question is modelling either expectancy, value or cost; with 3 expectancy, 3 values and 4 cost questions. The questions for this solution were adapted to change them from a student's learning context to a teacher's learning context. This process involved changing references from being in a class to teaching a class, and changing the framing of questions to work for both: teachers who currently teach computing but are struggling and teachers who do not currently teach computing. An example of two questions can be seen below

I think teaching my class Computing is/would be useful for my students.

I know I can learn the material to deliver Computing.

Questions adapted from Barron et al. (2017); Kosovich et al. (2015)

The full list of questions can be found in Appendix A

The second aspect of the process is the motivational report. To calculate the motivational scores (split into Expectancy of Success, Perceived Value and Cost) for a user, the average is taken for all of the questions in that section (Barron et al. 2017). Each average is then categorised as either High, Medium or Low. These categorisations are displayed to the user, and are also used to calculate the user's resource recommendation. This is displayed through a website page, with the option of adding an email report as well, as shown in Figure B.2b.

6.3.3 Resource Design

The main content for the solution is the set of resources, designed to target motivational challenges, identified and modelled under EVT. These resources are the core component to improving a user's motivation towards computing. Due to time constraints of this project the resources

developed are a small subset of what would be required for a full product, and act as a proof of concept for the underlying theory of this solution.

A number of design requirements were developed for the resources to ensure they were effective and appropriate, based on the overall solution requirements identified in section 6.2. Each resource should:

- Be accessible to a user with limited computing background or knowledge
- Be engaging and efficient when delivering the content, due to teachers limited time
- Be simple for a user to interact with and use
- Aim to tackle an specific motivational barrier
- Have a clear link to one of the identified challenges modelled using Expectancy Value Theory
- Contain link to relevant resources and literature
- Be based on the Scottish BGE Curriculum

6.4 Design Summary

Overall the solution design aims to start teachers on a learning journey in computing. The solution will model a teacher's specific motivation and provide a personalised journey through the site, ensuring that the correct areas of the teachers motivation are targeted, all modelled using Expectancy Value Theory. Resources on the site are designed to target specific motivational challenges, based on the identified pain-points that teachers experience. This motivational increase will be done by creating resources that increasing teachers perceived value and expectancy of success and lowering their perceived cost of computing.

7 | Implementation

7.1 Implementation Details

The website was implemented using the `Next.js React Framework`¹ to provide a simple framework for implementing the site. The site used `Tailwind CSS (V4)`² for styling, as it comes pre-packaged with `Next.js`. The tech stack was designed to be lightweight and easy to implement reducing the development time needed therefore allowing for more time to focus on the underlying theory and development of resources.

The site makes use of a number of pre-made UI components from `Flowbite React`³ to provide clean, simple and familiar UI for users. Alongside the pre-made components the site makes use of Reacts components functionality to define a number of custom components, allowing reuse on different pages of the site, making it more efficient to develop pages, and creating a more consistent experience for users.

The site mainly used client-side rendering and processing to reduce the complexity of the webapp. As there is no login functionality or major data storage on the site, there is no complex database or server-client communication/synchronisation needed. This simplistic client-server architecture simplifies the development of the site.

7.2 Development Process

The development process of the resource was split into weekly/fortnightly sprints, with progress made during each sprint along with feedback on that progress discussed during a weekly/fortnightly meeting. Due to issues around access and time regarding teachers as stakeholders, initial feedback and evaluation was given solely by my academic supervisor, using their knowledge of teachers requirements and interpretations of the resource from their position as an academic lead in a computing science school teachers association.

7.2.1 Sprint 1

The most complicated and implementation heavy section of the site was the implementation of the EVC survey and the data processing and dynamic generation required to create a motivation report for each user.

The form required a custom component to be designed for the scale input, which was done by heavily editing a pre-made slider to create a Likert scale, with appropriate labels and slider behaviour. This single component went through a number of iterations to ensure user friendly and accurate behaviour. The form was developed to ensure that each question was appropriately labelled as expectancy (of success), (perceived) value or cost as defined by subsection 6.3.2. Question order is randomised each time the user visits the page, to account for potential order bias.

¹<https://nextjs.org/>

²<https://tailwindcss.com/>

³<https://flowbite-react.com/>

The main challenge during implementation was the submission and generation logic. On submission the site first processes their survey results on the client generating a score and mapping (high, medium or low) alongside a recommendation (is improvement in this area needed) for each category. This data is then used to generate and display the motivation report to the user. There were many different implementations attempted for this process, until the final implementation was reached.

The first attempt was to generate a PDF for the user containing their motivation report. This PDF would be generated dynamically and sent to the user via email, where the user would download the PDF and use it as a reference point. This implementation was infeasible due to outdated packages, issues with dynamic rendering and client-server communication limitations. It was decided to use a different implementation to ensure the project continued to progress.

At the same time initial development of the resources started. The initial design of these consisted of animated videos, made with lively music and animations to engage the user. Initial progress was slow due to the challenges developing resources that contained relevant information, but in a way that teachers could understand and learn from. This was a key learning point during the project, that communicating computing as a computing student to users that have limited or no understanding of computing is a challenge. This meant the initial resource developed did not develop teachers understanding and only re-stated known or surface level information.

7.2.2 Sprint 2

This sprint involved implementing new logic for the motivational survey. This implementation was to send a dynamically generated email containing the motivation report to the user. This made use of some existing work from the previous implementations email logic. While this implementation worked, a user having to provide their email address to use the site was identified as a blocker to usage. It was decided that receiving an email notification should be an option, to allow users to extract the report from the site, but should be fully optional with the motivational report generated within the site itself, requiring no personal information.

Further development of resources was undertaken during this sprint, attempting to introduce a deeper level of understanding of the resources content to teachers, while still being accessible without a computing background. This balance again proved to be challenging and led to slow progress creating resources, as such there was not much to show during this sprint. It was decided at this point to create a set of four resources for the proof of concept, with two targeting perceived value and two targeting expectation of success, highlighted in Appendix C. A significantly larger amount of resources would be required for any final product using this theory.

During this sprint feedback was revived about the messaging of the site, which at this point was CS Motivate and used strong references to improving teachers low motivation. This messaging could be perceived as being hostile towards teachers with potential implications of laziness and lack of care, which was not intended. It was decided to do a full messaging rework of the site, with no mentions of motivation initially, and slowly building up the mentions of motivation and the underlying theory once teachers had started engaging with the site. Where messaging changes were not possible, such as the Motivational Survey, wording was changed to ensure teachers were aware that low motivation was not a negative reflection on themselves, but caused by to environmental factors. The biggest part of this messaging change was on the home page, changing the sites "branding" from Motivate/Motivation to Empower/Empowerment.

7.2.3 Sprint 3

During this sprint further messaging changes were made, mainly focusing on the links to literature in the site, both through information pop-ups dotted around the site, and the dedicated About section. Much of the information at this point had been taken from reports that had

been generated during the research phase of the project, which required an understanding and background of computing and computing education. These sections were changed to make them more accessible to a teacher without a background in or knowledge of computing, alongside providing links to external references and my own reports if they required additional detail.

7.2.4 Sprint 4

The final implementation of the motivational survey was developed during this sprint. Taking onboard feedback from previous iterations (of this design) that it should not require any personal information or require much interaction from the user. As such it was decided to put the motivational report on a dynamically generated page that the user was automatically navigated to after a successful submission. The page was created using information passed through the URL (generated by the form submission and redirection), as HTTP GET arguments. This allowed users to bookmark this URL and revisit their report easily, with the page generating their report again based on the provided URL, without having to save the page on the client or server. The final implementation of the motivational report can be seen in Figure 7.1.

During initial evaluation a number of issues were highlighted with the resources, their content and their style. It was highlighted that the videos moved at a very fast speed, and did not allow teachers time to try and understand the information displayed to them, they just moved on. While teachers can pause the videos, often this would require them to pause and then try and rewind, breaking the flow of the resource. Further to this while the animated text-based videos made it easier to iterate on the content of the videos (as opposed to a voice over), the background music and visuals were often overwhelming and distracting instead of engaging.

There was also an issue identified around the content of the resources, repeating an issue identified during earlier sprints where not much useful information was being given to the user, with the resources often just reworking already available information. This was often due to the challenges when communicating to a non-CS user, and also limited time to develop an understanding of what was already known and available to teachers.

It was clear from this feedback that a number of major design revisions were needed. Firstly the resource format was changed from video resources to slide-based resources, in order to allow the user to progress at their own speed, make it easier to navigate and return to the resource. Secondly the content of the resources were adjusted, to be less holistic and abstract and more focused and concrete. To demonstrate this as a proof of concept it was decided to focus on BGE First Level and only on expectation around understanding of the curriculum. The complete resource would contain resources for all challenges identified, targeted at each level of the BGE curriculum.

The proof of concept resource set focuses on explaining how the current curriculum works, by introducing teachers to the underlying framework of the curriculum, the ABC model (defined in subsection 2.1.1). This was done by creating two resources, the first one introduced the underlying theory/model being the curriculum, and the second one walking through a lesson plan highlighting how the different sections of the curriculum/model interact with the lesson.

This feedback came late in the project, and should have been spotted earlier, however due to project time constraints to deliver the website and resources in a short amount of time, there was a lack of planning and self evaluation performed during early development. This was a key moment of understanding why planning and evaluating development is important, especially when operating in an unfamiliar area, and such improvements were made when revising the resources.

In order to ensure these resources improved on the existing set, while still meeting the requirements identified two storyboard documents were created, to outline the content and design of

the new resources, and attempt to provide more direction and formal process than the previous resources had during design and creation.

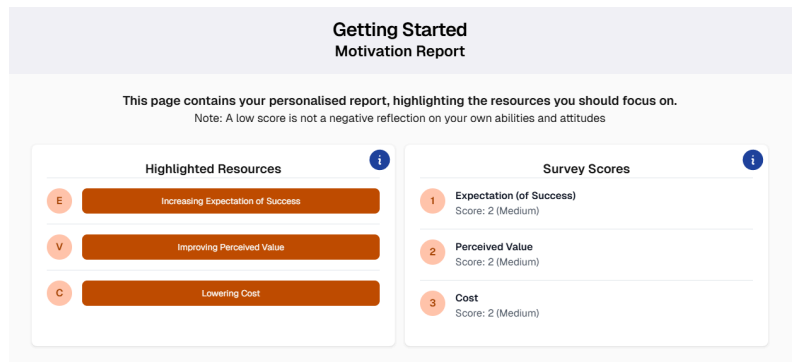


Figure 7.1: Screenshot showing the final motivational report page. This page can be dynamically re-generated from its URL.

8 | Evaluation

In order to evaluate the underlying theory and the realised proof-of-concept, a small-scale evaluation was conducted. This evaluation recruited a small number of participants who are involved in improvement of computing science education. While these participants will have a perspective on what does and doesn't work when engaging with teachers they do not represent the typical characteristics of a primary school teacher, particularly given all participants had a background in computing. The limited sample size of this evaluation restricts the generalisability of the results, providing only an initial understanding of how the underlying theory and realised proof of concept can be perceived.

Further larger scale evaluations are needed to gain an understanding of the effectiveness and guide future development of the theory and subsequent prototypes. Alongside this, future evaluations should directly involve primary school teachers as they are the primary stakeholders of this theory, in order to gain an understanding for their perspective and background.

8.1 Methodology

Before an evaluation session the participant was sent an information package containing an overview of the process and an introductory script. Participants were made aware of how to contact me to ask questions, and that they could withdraw at any time.

During the evaluation participants were first asked to complete an observed walk-through of the site, verbalising any thoughts, decisions or questions. This was recorded using both an audio recording and observation notes. Participants were then asked a number of questions in a semi-structured interview, to gain a deeper understanding of their thoughts and clarify anything that occurred during the walk-through. After the evaluation the participant was sent a debriefing script and provided an opportunity to ask any questions they had about the remaining process.

Each interview artifact (recording, observation notes) was uploaded into analysis software and coded using an inductive approach to refine codes, first on individual interviews and then considering interviews as a whole. These codes were then categorised into the three main aspects of the project: underlying theory, resource development and website design.

This evaluation design was chosen to attempt to replicate the experience that a teacher would have when viewing the resource for the first time, to understand how the design of the resources is perceived by a participant and to gain an understanding of how well the underlying theory is communicated to the participant through the site.

8.2 Results

8.2.1 Participants

Three participants were recruited via email invitation. While none of the participants were primary teachers they were identified as having relevant backgrounds in computing education

improvement. This allows them to provide initial feedback on the relevance of the underlying theory and the implementation of a proof of concept.

The participants had the following backgrounds: P1, P2 were fourth year students conducting projects in computing education with experience volunteering in primary schools. P3 was a former teacher currently involved in the development of computing education.

While these participants provide a useful initial perspective for the evaluation they are not representative of primary school teachers, specifically since all participants have a background in computing.

8.2.2 Underlying Theory

Relevance Participants agreed that teachers face a number of blockers when approaching computing, with two participants identifying a selection of the pain points identified in chapter 3 without being prompted by the researcher or the website. An example of this was one participant stating that teachers may feel that computing is too challenging to teach.

So they [teacher] might be apprehensive about the material might be like an anticipation might be too hard. (P3)

This independent identification of these pain points provides limited initial support that these pain points are relevant for teachers in Scotland, and that they might experience them when attempting to deliver computing and as such they should be core to the development of a solution.

One participant did however raise that while the pain points developed from the literature are relevant to Scottish primary teachers, they lack specific understanding of the situation in Scotland and as such it would be important for any continuation of this project to speak directly to teachers to develop an understanding of how the pain points specifically effect Scottish teachers.

The framing of the challenges teachers face as a motivational challenge and a learner journey was seen as a novel approach (than what has been used before for computing), highlighting the importance of understanding the effect that this method could have to develop in future studies. Participants also felt that EVT was a good model for teachers motivation, highlighting the core reasons why they may struggle to teach computing, especially when they are beginning to look at the subject. One participant's response to a question on the underlying theory was:

It's [EVT] targeting the kind of main issues that primary school teachers feel like as barriers for their computing teaching. (P2)

8.2.3 Resources

Relevance Participants felt that the existing resources focused on developing teachers' understanding of the curriculum providing the teacher with relevant information to help develop their knowledge of the curriculum and tackle that pain point. Specifically this was by introducing the underlying structure behind the curriculum, being the ABC model that describes a modelling process. Participants viewed this as relevant for both teachers trying to get into computing in order to understand that computing is not as daunting and complicated as they may have thought; and for existing teachers that deliver computing to understand what it actually is they should be teaching. This supports the resources' aim to increase teacher expectation of success by improving their understanding of how the curriculum works.

I guess if you're saying like when you do these activities, and you recognise them as ABC, they're strengthening your overall capacity kind of thing. (P3)

Participants highlighted how often students can use tools like Scratch but don't actually have an understanding of what they are doing, which the ABC model and the curriculum in general is designed to fix, developing understanding of all areas of the process.

And I feel like that's something I wouldn't have like picked up on at all that kind of like ABC aspects. I feel like lessons tend to be a lot more like practical focus and just on actually doing something rather than often developing any sense of understanding really. (P1)

More generally participants liked the use of non-computing examples to introduce concepts to ensure teachers understood it before framing it in a computing context. The two examples used in the Introduction to ABC resources, being Lego and literacy, were seen as examples that most teachers would be able to understand and related with, specifically the literacy example as it allows them to link computing to a subject area they understand and are teaching on an daily basis.

What does help them [teacher] is if they can see computing present in other areas of the curriculum (P3)

It was suggested that future resource sets should aim to use a consistent set of examples (both computing and non-computing) to make it easier for teachers to understand the underlying connections between the resources and what they are aiming to develop.

Bring back, have maybe two or three examples like Lego bricks ... then draw on them and keep drawing on them in the different resources here, draw on the same scenarios to demonstrate (P3)

Usage Looking at the design of the resources participants preferred the improved slide-style resources over the original video design, as was anticipated. This preference was mainly due to the ability for them to go through the content at their own speed, and be able to go back and forward between different slides, which participants typically did at least once when working through the resources.

Engagement The periodic use of quiz style questions throughout the Introduction to ABC resource was highlighted by participants as a key way of keeping teachers engaged with the resource and the site overall. It was suggested that a future implementation should develop the slide-style format to allow for user interaction on certain slides, especially around quizzes.

One participant spoke about adapting the resources to also allow teachers to use the resources in a group setting as well as individually, both as a way of ensuring teachers correctly understood the content of the resource, through group discussions; and to attempt to tackle the issue of isolation that teachers face around computing.

I think it probably would be more beneficial for people to do this with colleagues rather than on their own ... when they try to begin to explain to somebody else why they're saying what they're saying, that it becomes apparent what they mean. (P3)

Further to this it was highlighted that it would be beneficial to add discussion points at the end of each resource for teachers to consider and bring into discussions with their colleagues, engaging them in computing discussions outside of the resource.

A final suggestion to ensure engagement of primary teachers, given their limited time was to include suggestions for what resources a teacher could work through given a time estimate, to allow teachers to spend less time having to think about what to do, and more time actually engaging with the site.

It's just this thing about teachers not having enough time is that I like the idea of your resource. I've got 10 minutes, I've got 15 minutes. So what can you usefully get people, what's a useful chunk of thing that they can do so that they feel that was good. I'll come back and do a different 15 minutes at another time. (P3)

Limited Scope As expected participants noted that while the current resources were a good start there needs to be a significantly larger set of resources to fully target all of the pain points and motivational challenges identified. It was noted that a larger resource set may appear confusing with the current resource page layout, containing a long list. Future implementations should reconsider the design of the resource pages to explore splitting resources by motivational challenge not EVT aspect.

8.2.4 Website Design

User Experience Participants noted that the site was easy for them to use, with a clean and simple UI. All participants easily identified the main route through the site, navigating from the home page immediately to the getting started page. Making the site easy to use and navigate was a key requirement identified, and a similar approach should be taken for further prototypes.

There were some graphical bugs discovered by participants, due to different screen sizes from ones used during development. While this is expected from a proof-of-concept further testing should be carried out on future prototypes to ensure there is no unexpected UI behaviour.

Links to Theory While most participants immediately went to fill out the survey after navigating to the "Getting Started" page, one went to explore the literature links first, noting how they found the "About" section provided useful context. Participants would often click on the Information icons dotted around the site to get additional background and context. As not all participants engaged with the full "About" section, it suggests that the decision to keep this separate to not overload teachers with information that isn't directly required, was apparent.

I like how you've got the see research part of it. That's quite good actually. You can go in and actually look at that. (P1)

Lack of Description Some parts of the site were identified as being unclear for a user not familiar with the site or the underlying theory. A specific section that was identified was the Motivational Report page, with additional information suggested on what the scores and categorisation displayed mean and how the report should be saved and used.

And maybe useful just to have on this page as well a little bit of an extra explanation of what that actually like means ... have a mini scale or something that would show like what value would actually classify as. (P1)

Personalisation The motivational survey and report were identified as a positive way of engaging teachers using the site, providing them with an understanding of what is causing their challenges with computing. It was suggested that this area should be increased allowing for additional personalisation on the site, with a more detailed motivational report and the ability for users to select the challenges they face directly alongside using EVT to model their motivation. If a larger resource set was developed in a future implementation a more detailed report could be developed to directly link to resources instead of linking to the aspects of EVT.

I guess, maybe almost where you kind of ask them in a sense ... instead of just like telling teachers, maybe say something like, I don't know, do you ever feel like you just can't like cover all this material or get them to like a depth of understanding? .. A bit more like talking to them rather than about, maybe. (P1)

8.3 Discussion

While limited in scale the evaluation provided some initial evidence that viewing the challenges that teachers face as a motivational challenge in a learner context was perceived as relevant by the participants. While further evaluation is required to understand the effectiveness of this motivational approach, this provided some initial evidence to show the underlying theory has potential to be developed further.

The proof of concept developed for the theory was also positively perceived by participants specifically around its relevance to the issues teachers face and the design of the website and resources. Further development is needed to introduce interactive elements into resources and significantly expand the number of resources to cover all of the identified motivational challenges.

There is scope for the theory to be developed further with more in-depth understanding of the pain points given teachers views and the creation of larger prototypes. Primary teachers should be involved at multiple stages of future projects to enhance the design, as they are primary stakeholders.

9 | Conclusion

9.1 Summary

This dissertation investigated the challenges faced delivering primary school computing education, analysing the importance of the subject, curriculum structure and teachers' experience from computing education literature in order to develop a set of pain points for primary school computing education in Scotland. From these pain points teachers were identified as a common factor with specific issues around their understanding, value and confidence towards computing identified throughout these pain points.

A motivational problem was identified that needed to be tackled as the initial step to improving computing education. Re-framing the improvement of computing education as a learner journey for teachers and making use of a student motivational theory to model teachers motivation provides a different approach than seen current literature. This approach allows for the framing of each of the identified pain points as a motivational challenge through Expectancy Value Theory.

A proof of concept website containing a set of resources aimed at tackling these motivational challenges was produced, in order to demonstrate the validity of this approach to improving computing education could work. While the limited evaluation did not fully show the effectiveness of this theory, it does show the positive perception that the theory and proof-of-concept received. The evaluation provided insight into what could be done to further improve how the theory would be perceived by primary teachers. With additional work focused on expanding the theoretical framework and the application of this theory there is the potential to improve how teachers perceive and approach computing education, motivating them to explore the subject further.

This project demonstrated that framing teachers challenges to teaching computing as a learner journey and a motivational challenge has the potential to be expanded and used to improve computing education in primary school.

9.2 Reflection

Attempts to improve computing education are faced with a number of fundamental challenges, many of which I did not anticipate when I undertook this project. Primary school teacher's are faced with many barriers to teaching computing however there is a lack of research that evaluates and analyses these issues holistically in order to identify common themes. Most existing solutions only focus on improving a narrow section of computing education, with little focus on how this would impact the area as a whole. This meant that a significant amount of the projects time was spent researching and analysing existing literature on computing education in order to produce a list of pain points that teachers experience when they attempt to teach computing.

The amount of time spent analysing the core problem limited the amount of time that was available to develop and evaluate a proof of concept for the theory developed. This meant that both the proof of concept and evaluation were smaller than initially planned in order to ensure a high quality artifact was created. While this limits the ability to generalise the result of the

evaluation and provided no in situ evidence of the effectiveness of the theory, it did show that the theory was relevant and has potential to be expanded and developed further. Evaluation with practising primary teachers had been envisioned by the project supervisor as being possible however in the end due to the time constraints faced by the project this didn't fit well into primary teachers schedules.

9.3 Future work

A number of areas of this project would benefit from additional time and development in order to enhance the underlying theory and future application further.

User studies should be carried out on current primary school teachers in Scotland to gather evidence to support the stance of the literature used to develop the pain points, providing specific understanding of how Scottish teachers are affected by these challenges, in order to create a stronger foundation for the theory.

Alongside this, as this project was only a proof of concept and not a full solution a larger set of resources should be developed on top of the existing set. These should take into account the feedback received during evaluation to cover more of the identified motivational challenges that teacher face, and provide relevant resources for teachers in all levels of the BGE curriculum.

Larger scale evaluation studies should be run on any future work, to gain a better understanding of how the theory and application works in practice, targeting primary school teachers who are experiencing these motivational challenges and understanding how they interact with and are impacted by the theory and the produced application.

A | Expectancy Value Cost Survey Questions

Hamish Allan

Getting Started Survey - Adapted EVC Questions

Adapted Questions

Expectancy

1. I know I can learn the material to deliver Computing.
2. I believe that I can be successful teaching Computing.
3. I am confident that I can understand the material to teach Computing.

Value

1. I think teaching my class Computing is/would be important.
2. I value teaching my class Computing.
3. I think teaching my class Computing is/would be useful for my students.

Cost

1. My Computing preparation requires too much time.
2. Because of other things that I do, I don't have time to put into teaching Computing.
3. I'm unable to put in the time needed to do well when teaching Computing.
4. I have to give up too much to do well when teaching Computing.

Initial Questions

Expectancy

1. I know I can learn the material in my [SUBJECT] class.
2. I believe that I can be successful in my [SUBJECT] class.
3. I am confident that I can understand the material in my [SUBJECT] class.

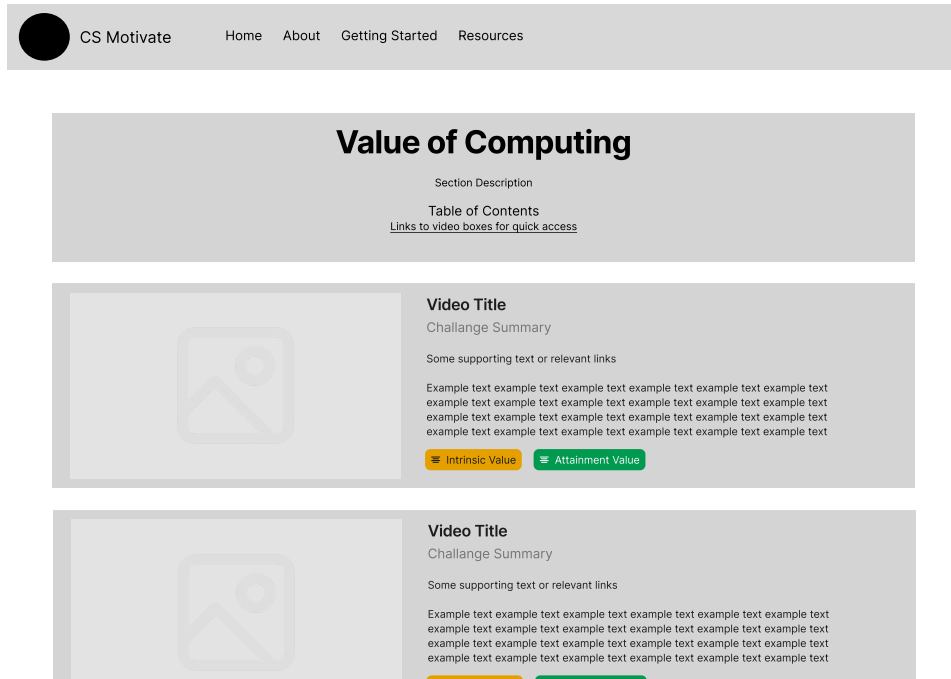
Value

1. I think my [SUBJECT] class is important.
2. I value my [SUBJECT] class.
3. I think my [SUBJECT] class is useful.

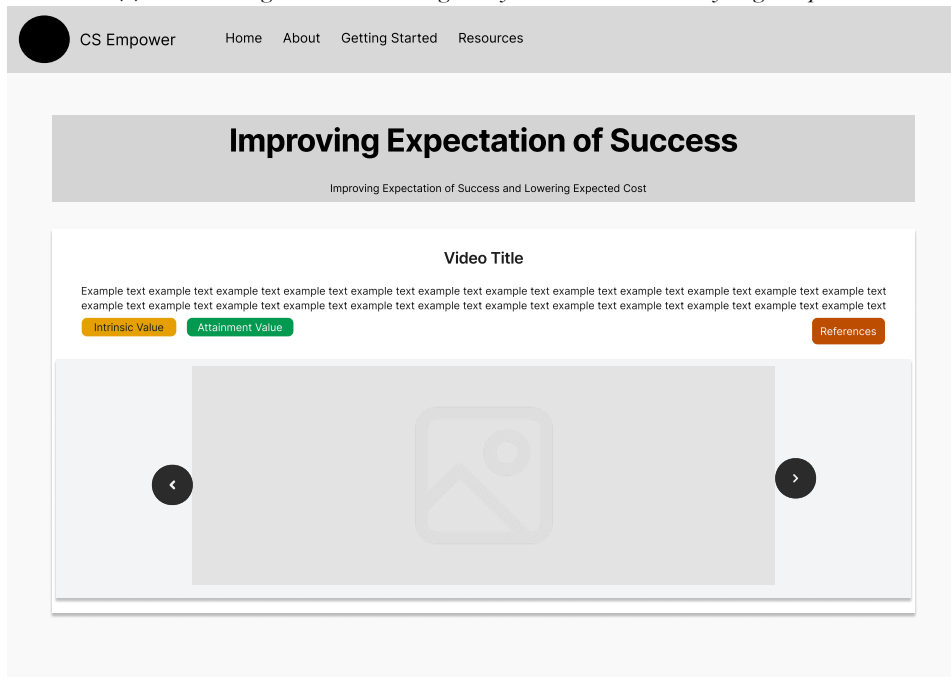
Cost

1. My [SUBJECT] classwork requires too much time.
2. Because of other things that I do, I don't have time to put into my [SUBJECT] class.
3. I'm unable to put in the time needed to do well in my [SUBJECT] class.
4. I have to give up too much to do well in my [SUBJECT] class.

B | Website Wireframes



(a) Resource Page Version 1 making use of video resources in a left-right layout



(b) Resource Page Version 2 making use of slide style resources in a top-bottom layout.

Figure B.1: Wireframes showing the different revisions of the resources pages for the site.

CS Empower Home About Getting Started Resources

Getting Started

Identifying Motivational Challenges & Blockers

Question Text

Strongly Disagree Disagree Slightly Disagree Slightly Agree Agree Strongly Agree

Question Text

Strongly Disagree Disagree Slightly Disagree Slightly Agree Agree Strongly Agree

Question Text

Strongly Disagree Disagree Slightly Disagree Slightly Agree Agree Strongly Agree

Consent Check
See Privacy for details

Evaluation Check
See Privacy for details

Submit Reset

(a) Motivational Survey Page. Using Expectancy Value Cost Survey to measure a users motivation

CS Empower Home About Getting Started Resources

Getting Started

Motivation Report

Resource Plan

Expectation (of Success)

Resource A Resource B Resource C

Perceived Value

Resource D Resource E Resource F

Cost

Resource G Resource H Resource I

Motivation Score

Expectation (of Success)
Score: 2 (Medium)

Perceived Value
Score: 1 (Low)

Cost
Score: 4 (High)

(b) Motivational Report Page. Using the result of the EVC survey to measure motivation and recommend resources

Figure B.2: Wireframes showing the pages required for Measuring and Reporting a user's motivation.

C | Video Resource Set

Table C.1: Table containing the description of each video resource developed for the site

Video Title	Video Description	Motivational Challenge	EVT Area(s)
Computing Skills: What are they?	Communicating the core skills that teaching computing develops in students	The computing curriculum is a small area with limited scope and importance. Teachers may not be aware of the skills that computing develops for their students	Value: Utility and Attainment
Computing Skills: Important skills for life	Communicating how important the core skills that teaching computing develops are for students outwith computing	Teachers may not be aware of the benefits of integrating computing with other subjects	Value: Utility and Attainment
Teaching Computing: What methods are there?	Communicating that there are many different methods for teaching computing, allowing a teacher to cater to their confidence and their classes' preferences	Teachers are cautious about computing. Computing requires a large amount of prep time, taking away from other subjects	Expectation: Lack of Subject Knowledge; Cost: Preparation
Computing Curriculum: What does it mean?	Explaining how the curriculum is structured and what each objective and outcome actually means	BGE Computing can be taught without needing extensive computing knowledge. Support and resources are available for the Scottish curriculum	Expectation: Lack of Subject Knowledge; Value: Attainment

D | Slide-Style Resource Set

Table D.1: Table containing the description of each slide-style resource developed for the site

Resource Title	Resource Description	EVT Area(s)
How does the Curriculum Work?	Explain what these three parts of the curriculum are, and how they interact with each other through the ABC Model	Expectation: Subject Knowledge;
ABC Model in Practice	The three organisers of the curriculum can all be developed through a singular lesson, walk through a STACS P2 (First Level) lesson plan, highlighting how the lesson works and how the lesson targets each part of the ABC Model	Expectation: Subject Knowledge; Expectation: Materials and Resources; Cost: Preparation; Value: Attainment

E | Evaluation Resources

E.1 Introductory Script



Level Four Project - Introductory Script

Improving Primary Teachers Motivation for Computing

Hamish Allan

The aim of this investigation is to get feedback on how well the created resource (motivate.hamishallan.uk) serves as a proof of concept to the idea of using motivational theory to improve primary teachers interaction with computing.

As part of this investigation you will:

1. Be observed working through the resource commenting on your experiences and decision making (~ 30 minutes)
2. Take part in a short interview on your experience with the resource (~ 30 minutes)

The site does not transmit your data unless explicitly granted. Any data shared will be anonymised and not shared outwith the University of Glasgow. No identifiable information will be collected unless you choose to disclose it.

You are free to withdraw from the evaluation at any time.

The evaluation is a test of the system, not your own abilities or motivation.

You will be asked to consent to take part in this evaluation when completing the getting started form. By agreeing to take part you are confirming

- Have read and understood this document
- You are over 18
- A Scottish teacher or similar
- Agreeing to take part in this evaluation

If you have any questions please feel free to contact me at 2792542A@student.gla.ac.uk

E.2 Interview Format



Level Four Project - Interview Template

Improving Primary Teachers Motivation for Computing

Hamish Allan

As part of the evaluation process for this project, a short 30-minute semi-structured interview is to be carried out with each participant. This interview can be either online or in-person. The interview will be recorded. This document is a rough outline of the interview, however questions will be adapted, added or removed based on the participants answers.

Introduction and Consent

Go through the introduction script with participants to ensure they understand the purpose of this evaluation. Ensure participants understand how this interview is going to work, and consent to take part and be recorded. Make clear that they can withdraw at anytime if they wish. There is no intensive or reward offered for participating in this evaluation.

1 Underlying Theory

Can you briefly describe what you believe to be the purpose of the theory and resource?

Do you feel like this approach is different or similar to others you have seen or experienced?

Do you feel that the theory is relevant for a primary school teacher?

2 Proof of Concept Resource

What do you think about the overall presentation of the site and individual resources?

What do you think about the resources links to the underlying theory?

Do you feel the resources would be engaging for a primary school teacher?

Do you feel the resources would be accessible for a primary school teacher with limited background in CS?

Given this is a proof of concept, what do you think would need to be changed or improved to make this a full resource?

Conclusion

Is there anything you want to add that did not fit into any of the above questions?

Do you have any questions for me?

Explain what the next part of the evaluation process is going to look like. They don't need to do anything else but their answers will be anonymised and analysed as part of the evaluation.

Make sure the participant understands how to contact me (2792542A@student.gla.ac.uk) and state again that they are able to withdraw at any time.

E.3 Debriefing Script



Level Four Project - Debriefing Script

Improving Primary Teachers Motivation for Computing

Hamish Allan

The aim of this investigation is to get feedback on how well the created resource (motivate.hamishallan.uk) serves as a proof of concept to the idea of using motivational theory to improve primary teachers interaction with computing.

If you have any questions please feel free to contact me at 2792542A@student.gla.ac.uk

Thank you for taking part!

Bibliography

- Alkaabi, A., Jabeen, F., Dwivedi, A. and Jose, S. (2025), 'Artificial intelligence adoption in the service industry: investigating the role of perceived usefulness and ease of use', *International Journal of Productivity and Performance Management* pp. 1–23.
URL: <https://www.emerald.com/ijppm/article/doi/10.1108/IJPPM-06-2024-0369/1324666/Artificial-intelligence-adoption-in-the-service>
- Arfé, B., Vardanega, T. and Ronconi, L. (2020), 'The effects of coding on children's planning and inhibition skills', *Computers and Education* **148**.
- Banerjee, S., Agarwal, A. and Singla, S. (2025), *LLMs Will Always Hallucinate, and We Need to Live with This*, Vol. 1554 LNNS, Springer Science and Business Media Deutschland GmbH, pp. 624–648.
URL: https://link.springer.com/10.1007/978-3-031-99965-9_39
- Barron, K. E., Taylor, J., Getty, S. G. and Hulleman, C. S. (2017), Expectancy-value-cost survey of student motivation: User guide., Technical report, The Motivate Lab.
URL: https://www.researchgate.net/publication/319433479_User's_Guide_for_the_Expectancy-Value-Cost_Survey_of_Student_Motivation
- Bartholomew, S., Yauney, J., Wuthrich, V., Wolfley, K., Elya, E., Rich, P., Shumway, S. and Wright, G. (2023), Digital storyboards: Making cs elementary, in '2023 Intermountain Engineering, Technology and Computing, IETC 2023', Institute of Electrical and Electronics Engineers Inc., pp. 232–237.
- Becker, B. A., Denny, P., Finnie-Ansley, J., Luxton-Reilly, A., Prather, J. and Santos, E. A. (2023), Programming is hard - or at least it used to be: Educational opportunities and challenges of ai code generation, in 'SIGCSE 2023 - Proceedings of the 54th ACM Technical Symposium on Computer Science Education', Vol. 1, Association for Computing Machinery, Inc, pp. 500–506.
- Bocconi, S., Chiocciariello, A., Kamylyis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M., Jasutė, E., Malagoli, C., Masiulionytė-Dagienė, V. and Stupurienė, G. (2022), State of play and practices from computing education reviewing computational thinking in compulsory education, Technical report, European Commission.
- Böckeler, B. (2025), 'How far can we push ai autonomy in code generation?', *MartinFowler* .
URL: <https://martinfowler.com/articles/pushing-ai-autonomy.html#CanFutureLLMsResolveTheseIssues>
- Caspersen, M. E., Diethelm, I., Gal-Ezer, J., McGettrick, A., Nardelli, E., Passey, D., Rován, B. and Webb, M. (2022), Informatics reference framework for school, Technical report, Informatics for All.
URL: <https://dl.acm.org/doi/book/10.1145/3592625>
- Connor, R., Cutts, Q. and Robertson, J. (2017), 'Keeping the machinery in computing education', *Communications of the ACM* **60**, 26–28.

- CSTA, CODE, ACM, CIC and NMSI (2016), K-12 computer science framework, Technical report, K-12 CS.
URL: <http://www.k12cs.org>.
- Cutts, Q. (2024), Orienting learners and teachers in introductory programming classes: The abc framework, in 'ACM International Conference Proceeding Series', Association for Computing Machinery, pp. 1–4.
- Cutts, Q., Robertson, J., Donaldson, P. and O'Donnell, L. (2017), 'An evaluation of a professional learning network for computer science teachers', *Computer Science Education* 27, 30–53.
URL: <https://www.tandfonline.com/doi/full/10.1080/08993408.2017.1315958>
- DfE (2025), Generative artificial intelligence (ai) in education, Technical report, Department for Education (DfE).
URL: <https://www.gov.uk/government/collections/using-ai-in-education-settings-support->
- Eccles, J. S. (1983), Expectancies, values, and academic behaviors, in 'Achievement and achievement motives', Freeman, pp. 75–146.
- Education Scotland (2017), Technologies: experiences and outcomes 2, Technical report, Education Scotland. Current aims of Digital IT Skills.
- Education Scotland (2021), Assessment within bge 2020/21 (update), Technical report, Education Scotland.
URL: <https://blogs.glowscotland.org.uk/ea/public/assessmentandmoderation2020/uploads/sites/16747/2021/04/21103421/Assessment-within-BGE-2020-21-Update.pdf>
- Elliot, A. J. and McGregor, H. A. (2001), 'A 2x2 achievement goal framework.', *Journal of Personality and Social Psychology* 80, 501–519.
- EuroStat (2025), '32.7% of eu people used generative ai tools in 2025'.
URL: <https://ec.europa.eu/eurostat/web/products-eurostat-news/w/ddn-20251216-3>
- Flórez, F. B., Casallas, R., Hernández, M., Reyes, A., Restrepo, S. and Danies, G. (2017), 'Changing a generation's way of thinking: Teaching computational thinking through programming', *Review of Educational Research* 87, 834–860.
URL: <https://journals.sagepub.com/doi/10.3102/0034654317710096>
- Gong, X., Xu, W. and Qiao, A. (2025), 'Exploring undergraduates' computational thinking and human-computer interaction patterns in generative progressive prompt-assisted programming learning', *International Journal of Educational Technology in Higher Education* 22.
URL: <https://educationaltechnologyjournal.springeropen.com/articles/10.1186/s41239-025-00552-y>
- Greifenstein, L., Graßl, I. and Fraser, G. (2021), Challenging but full of opportunities: Teachers' perspectives on programming in primary schools, in 'Proceedings of the 21st Koli Calling International Conference on Computing Education Research', ACM, pp. 1–10.
URL: <https://dl.acm.org/doi/10.1145/3488042.3488048>
- Kosovich, J. J., Hulleman, C. S., Barron, K. E. and Getty, S. (2015), 'A practical measure of student motivation', *The Journal of Early Adolescence* 35, 790–816.
URL: <https://journals.sagepub.com/doi/10.1177/0272431614556890>
- Larke, L. (2019), 'England's computing curriculum - a clash of values, beliefs and purpose - bera'.
URL: <https://www.bera.ac.uk/blog/englands-computing-curriculum-a-clash-of-values-beliefs-and-purpose>

- Learning Directorate (2012), Supporting Scotland's stem education and culture – science and engineering education advisory group – second report, Technical report, Scottish Government.
URL: <https://www.gov.scot/publications/supporting-scotlands-stem-education-culture-science-engineering-education-advisory-group/>
- Marcelino, M. J., Pessoa, T., Vieira, C., Salvador, T. and Mendes, A. J. (2018), 'Learning computational thinking and scratch at distance', *Computers in Human Behavior* **80**, 470–477.
- Mason, S. L. and Rich, P. J. (2019), Preparing elementary school teachers to teach computing, coding, and computational thinking, Technical report.
- NAP (2013), *Next generation science standards : for states, by states*, National Academies Press.
- Nardelli, E., Futschek, G., Gal-Ezer, J. and Webb, M. (2025), A European survey on requirements for the informatics school curriculum and informatics school teachers' education, Technical report, Informatics For All.
- OECD (2023), *OECD Digital Education Outlook 2023*, OECD.
URL: https://www.oecd.org/en/publications/oecd-digital-education-outlook-2023_c74f03de-en.html
- Ramirez-Salgado, A., Wusylko, C., Weisberg, L., Delgado, J. and Israel, M. (2025), 'Preparing elementary preservice teachers to integrate computational thinking in the curriculum', *ACM Transactions on Computing Education* **25**, 1–22.
URL: <https://dl.acm.org/doi/10.1145/3744751>
- Rich, P. J., Browning, S. F., Perkins, M. K., Shoop, T., Yoshikawa, E. and Belikov, O. M. (2019), 'Coding in k-8: International trends in teaching elementary/primary computing', *TechTrends* **63**, 311–329.
- Rich, P. J., Mason, S. L. and O'Leary, J. (2021), 'Measuring the effect of continuous professional development on elementary teachers' self-efficacy to teach coding and computational thinking', *Computers and Education* **168**.
- Salehi, S., Wang, K. D., Toorawa, R. and Wieman, C. (2020), Can majoring in computer science improve general problem-solving skills?, in 'SIGCSE 2020 - Proceedings of the 51st ACM Technical Symposium on Computer Science Education', pp. 156–161.
- Sentance, S. and Csizmadia, A. (2017), 'Computing in the curriculum: Challenges and strategies from a teacher's perspective', *Education and Information Technologies* **22**, 469–495.
- Sentance, S., Kirby, D., Quille, K., Cole, E., Crick, T. and Looker, N. (2022), Computing in school in the UK & Ireland: A comparative study, in 'Proceedings of the 2022 Conference on United Kingdom & Ireland Computing Education Research', ACM, pp. 1–7.
URL: <https://dl.acm.org/doi/10.1145/3555009.3555015>
- Society, R. (2017), *After the reboot : computing education in UK schools*, The Royal Society.
- STACS (2025), Stacs: Annual report 2024/25, Technical report, Scottish Teachers Advancing Computing Science.
- Suh, J., Lee, K. and Lee, J. (2025), 'Programming education with chatgpt: outcomes for beginners and intermediate students', *Education and Information Technologies* **30**, 19511–19536.
URL: <https://link.springer.com/article/10.1007/s10639-025-13542-4>
- Tinto, V. (1975), 'Dropout from higher education: A theoretical synthesis of recent research', *Review of Educational Research* **45**(1), 89–125.
URL: <https://doi.org/10.3102/00346543045001089>

- Tsui, A. (2023), 'The crisis with computing in primary schools'.
URL: <https://www.computingatschool.org.uk/forum-news-blogs/2023/february/the-crisis-with-computing-in-primary-schools/>
- Vegas, E., Hansen, M. and Fowler, B. (2021), Building skills for life: How to expand and improve computer science education around the world, Technical report, Brookings.
- Wigfield, A. and Eccles, J. S. (2000), 'Expectancy-value theory of achievement motivation', *Contemporary Educational Psychology* **25**, 68–81.
URL: <https://linkinghub.elsevier.com/retrieve/pii/S0361476X99910159>
- Wing, J. M. (2006), 'Computational thinking', *Communications of the ACM* **49**, 33–35.
URL: <https://dl.acm.org/doi/10.1145/1118178.1118215>
- World, D. (2025), 'Global study reveals 91of respondents use chatbots to conduct research _ digitalisation world'.
URL: <https://form.digitalisationworld.com/news/67338/global-study-reveals-91-of-respondents-use-chatbots-to-conduct-research>