Finding Chapter One - Missions

## Other Documents

This chapter builds towards a Matrix that could be included in a concluding chapter. https://docs.google.com/document/d/1DRgpgIUfq5XIktYy-lX4WlviUDnffNc0v8SOeLGgHho/edit Also the RQ questions which answered https://docs.google.com/document/d/1RZnznbR4HKrgL8VFfIqoCrCAJM0mf\_sM8iEN4KCGeZ4/edit And the coding structure here  
https://docs.google.com/document/d/19PafC\_w\_7uObYL4v86IGeeFZTw2yKxjHfP\_k85rHk\_s/edit#

## Introduction to This Writing

In the previous chapters we will have looked at the research landscape to develop coding and design practices in line with social understandings of learning. In particular, the intentional development of a communities of designers and coders with an educational aims has been explored paying particular attention strategies that educators can use to aid these processes.

While this piece of writing contains results, it also contains references to concepts and methodologies which will be included in other chapters in the final thesis. The following bullet pointed list gives an overview of an emerging chapter structure and contents.

* Introduction
  + Context
    - Coding digital making’s movement from informal -> formal settings / curriculum
    - Game making as a particular focus of digital making
  + Inclusive approaches and the value of Project Based Learning, Universal Design for Learning
    - Critique of implementations of Computational Thinking as a non-inclusive approach
    - Use of Game design patterns as a more “relatable” pedagogy
  + Description of Research Questions
  + Brief outline of thesis chapter structure
* Literature Review -
  + What game making pedagogies exist?
    - UMC and Games - Half-baked games
    - Design-based & Tinkering / Exploratory
    - More on Game design patterns - a shared language - turn to community aspect
  + What exists to support the development of game making communities?
    - Game making as a non-formal activity - defining terms informal/ formal / non-formal
    - Social learning in Game Making
      * Game playing and making communities
      * Games as funds of knowledge in
    - Visual methods in game making
      * Maps of learning, Gee, borrowing how to chart learner’s journey from game design
* Methodology
  + Summary of social turn
  + Justification of choice of methodology
  + Design-based research and Intervention studies within educational field
  + Data gathering methods and justification
  + journal, evolution of design, video analysis,
  + Data Processing, technical nod to challenges and resolutions
  + Data Analysis and justification
    - Analysis of non-video data, diary, notes, created Resources
      * Coding to 3 planes & other techniques
    - Analysis of video Data
      * Coding structures, design stage, 3 planes, game element addressed, participant interaction
  + Research Questions
* Chapter on Emerging Design
* emergence of focusing of Design Patterns - currently as different chapter
* Use of tensions as a useful analytical tool.
* Analysis of structure and evolution of design elements using 3 planes approach
  + cultural
  + social
  + personal
* Results Chapter - Cultural (inclusive ) Practices
  + Funds of Knowledge as they emerged - not universal, but pervasive
  + Game Player / Makers Types
* Results Chapter - Game Design Patterns
* focusing on Design Patterns
* Use of design patterns as a communicative tool
* Design patterns as a “middle level” construct to open door to more concrete and abstract
* Results Chapter - Personal Practices / Maps
  + Vignette on difficulty of X
  + Discussion linking to learning dimensions of Petrich and Bevan
* Conclusions and Next Steps
* Limitations of the study
  + The lack of motivation of ticking of curriculum led to less focus on individual knowledge work - This could be followed up
  + Other Limitation.
  + Interesting areas that have emerged.
  + Next areas of development.

**Comment:** More writing in this area is available online at - https://mickfuzz.github.io/phd4/thesis\_chapters.html OR just create hyperlinks.

## Context for this writing

To help the reader understand the data in this chapter the following notes on methods/ methodology are included.

* The study uses mixed methods in methodology. Informed by analysis of journal diary, notes, video data and
* An evolving table of what data is used in what way to answer each RQ is coming but is not ready.
* The research design evolved as the game making program was underway based on feedback and reactions of the participants (supported precedent in design-based research)
* A long pilot course and development stage resulted in extensive support materials and activities which I have structured into a learning design called 3M (more below)
* A narrow data samples has been used for this study. This prioritises a greater depth of analysis on interactions over generalisation. This is justified in a section on data validity.
* The concepts of resolving tensions within activity from Activity Theory are used in a previous chapter to explore the evolution of the designed
* Rather than Engestroms AT framework (Subject, Object, Instruments, Rules, Community, Division of Labour) the results chapters use Rogoff’s 3 planes of analysis (Cultural, Interpersonal and Personal)
* A quick summary of how parts of the design address three planes [is here](https://docs.google.com/document/d/1FIPn2dEB2aCKZi5z05j69cnHgpGNXFh1_SPV6h5l12g/edit?usp=sharing)

## Chapter overview

The working title of this chapter is Missions which aligns with a strand of the 3M game making pedagogical model developed in the course of this study.

The purpose of this chapter is to explore how the data gathered informs thinking on two research questions of this study.

* How can game design patterns support the development of coding practices by novices?
* How can the experience of fostering community focused digital game making practices inform wider practice in this area?

To do this, the chief analytical lens used in the chapter is that of interpersonal activity. Specifically in the form of guided participation in this game making community (Rogoff - explored previously).

This section starts with a Vignette of parent child interaction in game making process. The vignette is used to introduce an overview of interpersonal interactions of that pair via analysis of video data. The activity of the focal pair is then compared to activity of other groupings and individuals.

The chapter then focuses on the use of game design patterns (GDPs) by participants in these interactions and then in reference to wider cultural and personal planes of activity.

The working definition of GDPs explored in the literature review and enhanced in the design chapter is augmented with analysis of specific examples of their use by participants in practice.

### Vignette A

// create with google docs to markdown extensions // current version here - // then at this page online conversion. https://markdown-convert.com/en/tool/table // OR - AND PROBABLY BETTER TO DO THIS //keep as table with images - online or open office - then download and copy and paste into documents to be shared. // On line version is here - https://docs.google.com/document/d/1B7kFl5QsGV-oWtFWHLBIJNpa5GeYMlCntq31BnPAQT4/edit#heading=h.4rwirct7hqca

| **Dialogue** | **Activity** |
| --- | --- |
| S: You can ask him for help again if you want. | S is using laptop. T is nearby |
| T: Why |  |
| S: Cos I can’t do it. |  |
| T: Ok |  |
| Exchange follows with Mick and S - Problems solving using debugging methods and exchange of information about sequencing of variables. Ends after several minutes with game exhibiting desired behaviour of moving to next level after collecting a key. |  |
| M: Great | (On Screen) Platform game bug is resolved shown by game progressing to the next level. |
| S: Thank you. |  |
|  |  |
| Did you see T? | (Off Screen) M cedes computer to S. |
| T: Did it do it? |  |
| S: Yes |  |
| S: I’ll reload it. Here you go | S presses refresh key to reload the game |
| T: Laughs | T moves laptop to point towards her and takes over using the keyboard |
|  |  |
| T: (Makes excited cat noise) | T is using arrow keys to control game and check desired behaviour of player progressing to level 2 is working. T progresses |
|  |  |
| T: We need a door in this one. |  |
|  |  |
| I want to make more levels now. |  |
|  |  |
| So you can put lots of doors and lots of keys |  |
|  |  |
| Key Door Key Door Key Door |  |
|  |  |
| Key Door Level. Key Door Level, Key Door Level | S moves computer back to take over the use of they keyboard. She changes tab and navigates to the section of the code which deals with the design of levels and placement of different game components. |
|  |  |
| S. Well. Where are you going to put your door in level two? |  |
|  |  |
| Do you want to change the layout at all? | S gestures to screen |
| T: What’s door? | S doesn’t move laptop but gets closer to take over keyboard |
| S: d |  |
| T: Ok |  |
| T: d . Door right there. | Inserts a “d” in an array high above a platform |
|  |  |
| S: And where’s the key going? |  |
|  |  |
| Must be quite hard through get to that door |  |
|  |  |
| Key is K |  |
| T : (Calmly) Ok just let me do something | T starts to delete multiple characters on a line |
| S: (Alarmed) What are you doing? |  |
| T: You’ll see! |  |
|  |  |
| T: Ah yes |  |
|  |  |
| S: What are you.. Ah you’re making a platform. I see. | T starts to add x’s to replace the spaces she has deleted |
|  |  |
| T: See! |  |
| S: So where is the key going? |  |
| T: Key, so it’s K? |  |
| S: Yes |  |
| T: Where’s K gone |  |
| S: There | S points to the K Key on Keyboard |
| T: Back. K for Key |  |
| S: Are you putting one in your third level as well? |  |
| T: Yes and then I’m going to make more levels |  |
| T: You do this bit |  |
| T: And then delete those H’s. And in there. |  |
| S: No. No. |  |
| T: Then, let me do it. | (off screen) T takes over the keyboard |
| S: It’s no fun having a game with any hazards to avoid |  |
| T: Is for me! do you go that way back? |  |
|  |  |
| So sorry for deleting… They just… | (on screen) T deletes hazards in Level 3 of game |
| Back Back Back |  |
|  |  |
| Oh Oh Oh. |  |
|  |  |
| (laughs) |  |
|  |  |
| (replaces another) Back Oh (giggles) | (on screen) T adds three coins to Level 3 |
| S: Goodness me what what the point of designing our car with fumes if we’re not going to use it? |  |
| T: I don’t know (sighs) |  |
|  |  |
| T: You put the key and door in this one. |  |
|  |  |
| T: Go on then. Key - Door - Person. |  |
| S: Person? |  |
| T: Key Door Person. | (off screen) T indicates that her mother is the person she is referring to |
| T: (laughs) | T start play through of their game again. She progresses past the first level, completes second level. The game then goes back to the first level unexpectedly. |
| S: Is this the first level again |  |
| T: I think so |  |
| S: OK so our the door for the second level goes back to first level and we want it to go to the third level presumably |  |
| T: Yes (laughs) Yes. Why could that be? That’s silly |  |
| S: I’m just going to go see something. | S dives into the code to try to troubleshoot the problem. T watches for a short while becomes disengaged. She then leaves mother to explore the wider environment of the room and objects within it. |

#### Patterns of Activity in Vignette A

I coded two complete sessions of this parent child pair. The following observations have been made by comparing the codes on stages of design and types of participant interaction. This data was then triangulated with the accompanying transcribed fragments of conversation and my own journal notes. The observations are presented themed by the design stages with some general observations listed at the start.

**General Observations at all stages**

* This parent-child pair alternate between use of keyboard and mouse. Currently the parent takes the driving seat. (Their roles can be contrasted to pair programming).
* When parent problem solving the child is sometimes physically inactive, looking into space, but then explores the space and/or engages in peripheral activity with other groups (cf Rogoff, observation as valid form of participation)
* When possible, the parent engages child to input the design activities. This is done at different levels at different design stages.

**Ask / Imagine**

* In early ask/imagine stages parent and child discuss storytelling, graphical and other aesthetic choices

**Plan**

* in planning stages the parent provides scaffolding to help the child.
* In one example the parent notices the child’s difficulty in using cursor and delete/backspace keys to edit a matrix allowing level design. The parent provides a book with grid paper to allow the child to replicate the matrix. The parent then transcribes the design while engaging the child by checking she has interpreted the design correctly.

**Create**

* At certain points in the create stage the parent needs total concentration and is not able to respond effectively to requests of attention.
* At times the parent asked the child to seek help from facilitator.
* At another time, when the child was bored waiting for parent debugging, she approached the facilitator to ask for help on behalf of the adult without prompting.

**Test**

* The parent cedes the keyboard and mouse and sometimes swaps chairs to allow child to play test the game created.

**Improve**

* Parent navigates to a particular part of the code for child to alter design
* Child takes on challenge of making smaller changes

**Share**

* One of the most active times for the child allowing interaction to explain their game to others and to play the games of others.

#### Representing Coded the Patterns of Activity in Vignette A

Background: Recap of the coding structure decided upon. See current table here. https://docs.google.com/document/d/19PafC\_w\_7uObYL4v86IGeeFZTw2yKxjHfP\_k85rHk\_s/edit?usp=sharing

* Design Stages
* Interaction Description
* Game Elements Addressed

Here I will use Nvivo to give insight about the engagement of participants by compare time on different tasks.

The coding structure (may) also allows to show the overlap between different code areas. This is useful to explore a question like the following:

* Question One: During which design phase does parent-child pair interaction happen most frequently?
* Question Two: What are the different patterns for participant interaction that happen when working on different game elements?

TO COME.

The first scenario represented in a chart .

CHART

The second scenario is represented by 4 charts one for each game element showing the duration spent by a diversity of participants.

CHART

Followed by discussion

### Further observations and discussion on game making activity

**Note:** I am not totally clear on the purpose of this section. Needs revisiting in terms of chapter structure.

#### Comparing the activity of this family to others

We can compare this pairs pattern of activity with other pairs / families.

* The parent here is much more commonly rooted to the computer than other parents.
* This meant they engaged far less in play testing of other games and found it harder to gain attention for direct support.
* This grouping sometimes left early having completed more in code development than others but completed less social activities.

Discussion

* The parent had indicated that she considered herself a “planner” as a maker type which is confirmed by less social Interactions

The observations above show the importance of recording the whole room. Some participants will roam to observe the work of others, to socialise, to gain attention of others for support or for feedback.

When analysing data using 360 recorded video side-by-side with the screen captured data, the participant can be followed around the room and their activity can be noted even when away from their activity.

## Identifying GDP as key unit of Analysis

Note: How much of this needs to be relayed earlier in the Design chapter to explain design decision?

### Relating to AT and design Theory

CONTEXT PROBABLY PARTLY COVERED IN LITERATURE REVIEW:

Design patterns are most commonly used for computing students at higher education to teach object oriented computing but they are also useful for all levels of learners. Design patterns are rooted in real-life incidences of problems that are often solved in a particular way. They are concrete examples of coding principles in context. Design patterns can help the development of coding communities if more experiences coders take the time to document the patterns they use in an accessible way for novice coders. For educators the use of design patterns can help support learners develop coding proficiency by providing scaffolding and modelling good design decisions. However, one of the challenges for teachers of using worked examples and design patterns is how to integrate them into student-led design challenges.

* Drawing on AT terminology
* Discuss how GDP as key unit was arrived at - draw on journal notes in planning stages
* Relate - AT terms to those of design stages
* Begin to comment on utility of GDP as a unit of analysis for practitioners, parents and young participants - i.e. in navigation and goal setting.

#### Importance of Game Design Patterns in Vignette A

Note: it seems a bit strange to return to Vignette A at this point. Can the writing be structured to avoid this disconnect?

For both participants there is an explicit awareness of game making patterns and concept of the game as a dynamic system. Both seem to drive their work on the design and coding mechanics of the game.

Game making patterns are outlined by both the child and parent. They talk of adding more levels, of the mechanic of keys and doors and then progressing to another level. The child takes delight in outlining this pattern of game progression. The transformation from the period of time when her parent was problem solving code is dramatic. I propose that a contributing factor to this motivation is the ability to replicate a familiar pattern of game play via her own work in construction. It is the ability to translate this home knowledge into her own artefact which gives a sense of mastery over an otherwise foreign environment.

In terms of understanding of the game as a dynamic system, this is seen clearly in the parent’s alarm at the child’s deletion of all elements of hazard. The parent is keen to keep a sense of game balance to ensure a sense of challenge for the imagined player. “It’s no fun having a game without any hazards to avoid.” The child seems determined to remove all hazards. My understanding is that she is also aware of implications for game balance but is taking pleasure in this seeming destruction of the key challenge of the game as an act of disruptive play.

### Other Notes on use of GDP (to integrate with above)

* Parent offers choices of direction of next GDP
* The choice of a new GDP often involves new code to be added to the game or a significant change in code which needed debugging.
* For this pair, larger coding activities were beyond the ability of the child and the parent was not able to keep them engaged while solving the problem.

## Other Findings Emerging related to Game Design Patterns

Previously we have explored the concept of activity as a primary unit for analysis. In particular the utility of this approach to research community activity (Cite Blunden). In this section I will explore how game design patterns became the primary unit of activity for participants.

Discussion points

* How the GDP are sub activities goals in AT terminology.
* Thu, the game pattern goal rather than the overall making of a game, took up the majority of their focus.
* In playtesting with others the wider focus of making an engaging game comes into focus.
* The difference in goals in terminology
* How this unit allows a more detailed focus on some of the tensions involved in tool use

We have also looked at the other studies which look at game design patterns as a useful tool for advanced and less frequently for notice coders.

In the following sections, I draw on observations of participants in practice to outline how game design patterns are used in the making of games. I use extracts/ vignettes to do this.

### Cultural, Interpersonal & Personal Planes Analysis of use of GDPs

While each of these observations do not fall nearly into any one of the three planes being used as analytical tool, I begin with those oriented more to cultural focus which will deepen the observations in the previous chapter.  
and progress to the bulk of observations oriented chiefly to interpersonal activity and end with those more to do with personal knowledge construction.

## Cultural Plane / Apprenticeship

* Design patterns imagined used in other contexts as a form of apprenticeship
* Tools: Specifics on Code Playgrounds and Phaser and a Javascript framework
* the media literacy practices of transfer between tools - similar to other studies - see guided participation papers

## Interpersonal Plane (Guided Apprenticeship)

I will recap Rogoff’s take on guided apprenticeship and the features of the learning design which facilitate it.

If needed then I will explain how AT terms like division of labour are appropriate here.

### GDPs used as an organising tool

#### Task delegation

While there was no formal organising of groups in phase 2 and 3, the family bonds between parents and children created similar patterns of seeking validation, sharing planning. Although this was more the case for dyads working side-by-side on the same game, similar patterns existed for other configurations.

The opening vignette shows the name of a GDP as a way to communicate about the shared work of making a game. The utterance by the child “Key Door Person” seemingly indicated that the child ceded responsibility to the adult. The child has judged it the level of complexity needed to add a new pattern into the code to be beyond her ability and delegates appropriately.

|T: Go on then. Key - Door - Person.| | |S: Person?| | |T: Key Door Person.|(off screen) T indicates that her mother is the person she is referring to|

Gesture analysis (explained in a the methodology chapter) is valuable to shed light on this exchange. It is the indication that the mother is the person who will be responsible for implementing the GDP that the child is tangentially prompting to be implemented.

### GDPs and division of Labour

Similarly, rather than direct delegation, dividing of labour can be observed.

The implementation of some GDP involved the use of different tools and activities. For example, the design of a graphical element suited some learners and they depended on the support of others to implement those changes in code.

While the above interaction between Fi and Ma points to the parent doing the heavy lifting for code implementation, this was not always the case. Mi and Ne had a reverse relationship where the child would reluctantly implement the graphical changes in code but would pointedly not explain them seemingly taking pride in knowing something that her mother didn’t.

ADD TRANSCRIPT EXAMPLE FROM - 49:52.2 - 53:38.0 - 2019-03-18-mi

#### GDPs for Goal Setting and Prioritisation

The following exchange between participants Fi and Ma shows GDPs being used to organise future activity.

Online version as table - https://docs.google.com/document/d/1wLx8j9djYqJdZHwKJ09mmIDMYO7YqlMDhuHx2hiqpJI/edit

M: I’ve brought the music, and also we could just concentrate on one thing and just change that. You know, keep working through.

F: Yeah. I think I want to get an enemy in - oh no - my person animated.

M: so you want to get your person animated that’s the main thing

M: Shall we concentrate on that and changing the platforms into something different?

F: Yeah. M: Yeah? F: I also want to make a theme tune.

M: Yeah it’s, that’s what I mean, you can’t just skip around like that. F: hmmm M: just cos it gets really overwhelming. M: Yeah..? So…?

Long pause

M: Well I’ll have a look at the code and see if I can make sense of that.

M: I’ve brought the music, and also we could just concentrate on one thing and just change that. You know, keep working through.

F: Yeah. I think I want to get an enemy in - oh no - my person animated.

M: So you want to get your person animated that’s your main thing. F: Yeah

M: Shall we concentrate on that and changing the platforms into something different?

F: Yeah. M: Yeah? F: I also want to make a theme tune.

M: Yeah it’s, that’s what I mean, you can’t just skip around like that. F: hmmm M: just cos it gets really overwhelming. M: Yeah..? So…?

Long pause

M: Well I’ll have a look at the code and see if I can make sense of that.

This example shows the use of the names number of game design patterns by the child . While the planning discussion happens the child is concurrently testing their game. At this stage of the design some have been imagined and perhaps partly worked on and dis-guarded. In addition, the child describes a feature that they want to see happen.

This interchange shows a tension between a more chaotic style of working jumping from one goal to another and potential resolution with parental dialogue prioritising work to be done.

Here the child may be picking activities to fit their strong suit of creating graphical and audio assets in other gui based environments. Leaving the father to piece things together in the code based progam.

Or perhaps the parent steers the child into graphical areas as they are able to continue without stressing out the parent. Thus the child’s compliance may be a kind or pragmatic decision to allow smoother progress.

Discussion: The child’s initial listing of features is a brainstorming technique. Such techniques are used to aid a creative process however the parent seems to lack the skill to map these out and then to work together to priortise them. Instead he appears to be keen to quickly pick one that seems manageable.

GDP are already being used as a way to set goals, track progress and prioritise but the process seems stressful or unsatisfactory. Some kind of support help scaffold this process would be potentially productive.

Note: Such support is explored in more detail in the next chapter.

### GDP and sustaining engagement

The process of adding in a new GDP to the game was often a sustaining activity.

When GDP was completed and participant or pair move to a new phase there is evidence of a reenergising effect. e.g Example of adding a new level in 2019-03-18-th\_sh -

In the process of doing that, the pair return to a paper stage, and the process of turning that into reality on the screen and sharing with others is transformative in terms of energy level and interaction during the course of the session.

#### GDPs and personal expression (cultural)

Especially if part of the design allowed the expression of identity or a designed element - for example graphics, audio or a written entry.

Graphics –> cultural linking with home

Example of dog background graphics in 2019-03-18-th\_sh - The incorporation of the graphic was relatively quick and easy change to make the game but had a significant factor in the affect towards the game [erikssonUsingGameplayDesign2019;] - cultivativing a sense of ownsership - playing around the graphic - covering it over and revealing it - participant self-expression and a conversational prompt to other during sharing of the game

This is supported by other research on digital making and motivation. CITATION HERE? - or in cultural discussion?

Discussion on how cultural becomes an interpersonal aid through mediation? It allows young person to talk about dogs and their love of dogs. Even at one point that they used to have two and one died.

#### GDPs propagation through peer activity

The process of adding levels was something that was spread predominately through one child’s work to help others to add that features. The work of the child to add 21 levels to their own game served as a way to publicise this possibility. The role of that child as an expert in that design pattern, and their willingness to share their knowledge, shifted focus away from help from myself as a facilitator, or the instruction-based support documents.

PERHAPS ADD TRANSCRIPT EXAMPLE ?-

### Interaction of GDPs with Playtesting as method

Playtesting but in particular designing for others .

What was noticeable about - this in particular?

Playtesing as a regular practice is a immediate way to shift learners into a perspective designing for others.

This fits into our design cycle, as our first stages of ask and imagine are more productive when informed by a realistic sense of what the end user experience is. Regular self testing and playtesting achieve this.

Some examples: (should be many)

Pull on examples of learners imagining the experience of others. - Th - in session 2019-03-11 - comments on wanting students to be frustrated when playing it, it’s fun - then the final level being a secret fun experience playing against the norms of platform game design, thus illiciting surprise. - Another example -

Discussion - Designing for others - as a key perspective in constructionism. Note: What added insights or tips for practitioners can I bring here?

Also noticeable is the variety of ways in which playtesting happens and how younger and older participants looked to give and take from the experience.

Older participants often used this as a way of being supportive to both parents and young people. This happened in the following ways: - praising graphical content - making links with home interests of participants and building rapport - ( e.g. do you like dogs? )

### GDPs to support Iterative Design Practices

Iterative agile design is a highly cultural / interpersonal process.

Discussion drawing on more radical attempts to describe the best way to bring young learners into a design practice.

Perhaps in the field of participatory design.

Highlight how the granular and relatable nature of GDP

Well suited to build a repeated design practice. Especially with starting documents, Small changes had big effects. Variable changes needed immediate testing.

## Personal Plane

The personal plane is explored in a following chapter (working title Maps). This will cover findings relating to the value of participatory methods of mapping project progress and potential learning dimensions of activities.

### Mutual approaches to knowledge building

In this section the processes of personal appropriation of experience and knowledge emering from game making are examined closely in relation to particpants experience of GDPs.

* quick summary of Rogoff’s take on personal appropriation compared to a transmission model of knowledge & that participants re-contribute their experience of knowledge beck into the social mix
* e.g - GDPs used in this research program were mutually created based on feedback from participants
* While there is a curricular value attached to CT concepts & to some extent design practices, in this context participants can create their own understandings of what is useful to them.

### Game Design Patterns vs CT and more established “knowledge”

My end design highlights game patterns as a starting point for user interest and to draw participants in to other elements of learning dimensions. More abstract dimensions like systems thinking, and more concrete like coding concepts.

* Compare formal teaching of CT and CS with an approach based on a freer choice of patterns.
* Discuss the use of gdps as a gateway to other concepts as explored in CS in a more abstract, and more concrete direction (**???**) .
* Leading with GDP as a way to explore other Learning Dimensions
* Discussion - The use of GDPs hits a sweet spot as a “middle” level construct, (see Eriksson and Bjork)
* Discussion - Rising to the concrete - as a key AT concept - how does this fit?
* Make Code cards which contained links to game design patterns and the different component concepts
* (see Eriksson and Bjork)
* Draw on material and critique in chapter on semantic profiles / waves.

### Discussion on UMC and PRIMM in relation to use of Game Design Patterns? INCLUDE OR NEXT CHAPTER?

* UMC - should be covered in the Remix part of the design
* PRIMM - stage based - more prescriptive - probably suits next chapter.

## Incorporating findings into a pedagogical model for participants

Note: I’m not yet sure how much of the 3M model will be described in the “Results-Design” chapter. A short recap here seems appropriate.

As previously discussed, the process of developing the learning design happened in tandem with the data gathering. The model was guided by reflection and feedback from ongoing game making sessions. The following sections shift perspective slightly from a research perspective which seeks to inform research in the area of collaborative game making to an academic audience to a framing which will be accessible to practitioners and participants as well.

Note: To do this some areas of context are outlined which may end up in the opening chapters.

### Presenting GDP to participants as “Missions”

Many commercial open world games offer a central challenge consisting of small incremental missions and then optional side missions. Open world games increase user choice and thus give players a greater feeling of agency.

To mirror this approach, the main challenge of the 3M model is to create a playable game around a theme for a real or imagined audience with learners given the choice to add many optional features to the game. This approach steers students towards developing their use and understanding of coding structures, debugging practices and systems concepts. In addition, side missions encourage social and playful coding approaches which help develop a community of coders.

Note: TO ADD - The purpose of mirroring a game experience is to create a playful environment - perhaps already covered in previous chapter

THIS CONTEXT PROBABLY COVERED IN RESULTS-DESIGN CHAPTER: In the 3M model rather than following a step-by-step tutorial learners start with an incomplete game template and add new features as they choose. Each feature is described as a mission. This approach follows the Use-Modify-Create model to limit learner anxiety for novice coders and to scaffold the acquisition of coding and computational thinking concepts (**???**). I worked with learners to create a wish list of game features to create a 2D platform game. These features included moving hazards, jumping on enemies, finding a door or flag to progress to the next level. We can describe these features as game design patterns. Driven by the requests of learners, I developed tutorials to support students implement these patterns. This approach aligns with inclusive education principles in that it increases the choices of students, scaffolds the way they can access resources and allows them to keep a track of their own progress.

In my final implementation of the 3M model students picked missions from a choice of printed cards. There were four colour themed categories of missions. *Game mechanics* are features to do with the actions of the game. *Game space* patterns address the layout of the game. *Game polish* patterns involve adding music, backgrounds, graphics and story elements. Finally *System and Challenge* patterns look at how different elements interact to create challenge in the game. An example of a game mechanic design pattern follows.

PRINTED CARD TEXT BEGINS

Your mission is to apply the following pattern to your game. There are supporting step-by-step resources available if you need them. When you finish be sure to reflect on how adding this pattern helps your understanding of the computing concepts and similar patterns listed. This concludes your mission.

* **Name of Game Design Pattern:** Jumping on Enemies to Zap them
* **Type Pattern:** Game Mechanic
* **Description:** If the player is descending from a jump when they touch the enemy the player is zapped and in this case disappears.
* **Need for Pattern:** Enemies create challenge and being able to jump on an enemy is a good way of clearing the area you want to explore.
* **Coding Concepts involved:** Arrays[^8], Change Listener[^9]
* **Links to other Computing Patterns:** Systems Dynamics[^10]
* **Related Game Patterns:** You’ll need to have added the **Add Enemies** pattern to your game before you can add this one.

PRINTED CARD TEXT ENDS

In addition to outlines of game patterns, print-outs or on-line documents to support learners to implement the code needed are provided. While on-line documents allow learners to copy and paste code thus avoid many syntax errors, printed or incomplete code examples provide a greater level of challenge. Supporting resources help resolve tensions around learners getting stuck and needing a lot of facilitator help. These resources can help teachers deliver game making in a classroom context. Educators can alter resources to vary how much detail is provided in supporting documents to suit the challenge level for students. I work with young coders, thus I normally provide significant coding scaffolding. Once learners have built familiarity with code structures, processes and the coding environment, I provide less complete code examples and thus reduce the scaffolding.

## Additional Chapter Discussion

### Discussing Game design patterns as a tool to expose and give participants a language to discuss tensions

Synthesise examples which highlight the value of developing share language and short cuts. Draw from the above examples.

However, also look at potential short comings of short cuts - hiding more complex patterns within the name. When short cuts or misapprehensions prove less useful.

For example the shortcut of a name for a pattern in Fi and Ma’s interaction hides a large amount of problems solving which seems “overwhelming”

Is there an advantage to less support in terms of setting out a pathway which participants cover step by step. Or is the process of working on a sprawling problem which seems unmanagable, but which in with participants have to

### Design patterns as a “middle level” construct to open door to more concrete and abstract

In the introduction of this thesis I highlighted some of the challenges for inclusivity and motivation of more abstract approaches including a perscriptive approach to Computational Thinking.

Papert and Turkle celebrated diversity in approaches and celebrated the concrete.

In the relevant literature () - design patterns are seen as a useful middle ground between abstract CT concepts and more concrete techniques. There are the advantages of both a relatable, tangible shape of unit of activity linked to a community of practice. They also include a more abstract / and thus repeatable structure thus fulfilling some of the promise transfer offered by CT advocates.

The heart of the matter here is how transferable knowledge is without a contextual meaning attached. This is a contested area.

In this context of novice coders it seems appropriate to introduce some levels of reusable practice whilst prioritising individual learner engagement and community participation.

We can see evidence of both the potential for reusable elements of knowledge and practice involving practice in the following ways;

* Use of levels as a array which is parsed by a loop to create layout
* Other examples.

<Bibliography: Do Zotero Refresh>