Project Analysis

## Background to/identification of problem

I am intending to create a revision tool for A level maths . The program mainly be based around flashcards , and will generate questions with a difficulty based on how you found that topic previously as well as preset flashcards. My program will provide these questions in various A-level maths topics, and also provide the tools that are used to solve questions as an option on the main menu. Due to having randomised questions my program will also be useful for practising from the topic, as each question will be different and ones that were difficult could reappear again until you know how to solve them well.

She wants all these resources to be in one application to make it easier to revise.

## Description of the current system

Physical flashcards

* Requires lots of time for the user to make, which reduces efficiency.
* Offers no progression based on how difficult you find the cards.
* Has no was to easily replicate cards, without spending more time making them.
* Simple to use, and make.

Brainscape

* Offers Online flashcard customisation with some learning built into the card order via a user rating after completing a card.
* Works across multiple devices.
* Allows you to see how you're friends using the same deck are doing.
* Offers deck sharing to other users.
* Decks can be allocated to subjects, and be in multiple ones.
* Apart from sharing and using premade general decks creating the resources takes a lot of time.
* There's no premade random decks that get harder as you improve.
* It can choose the same card a lot if you still find it hard which is annoying if you'd rather see the other cards in your deck.
* Creating decks can take up a lot of time.

Book questions/Past paper questions

* Contains questions on only relevant topics.
* Only has a limited amount of questions.
* Requires time to find a suitable question of the right difficulty.

## Identification of the prospective user

My user will be Amy Packwood who's one of my friends currently doing A2 maths, and currently revises using a combination of practice questions ,flash cards and past papers. She hopes for the first two to become encompassed into one program, to make her revision more efficient.

## Identification of user needs and acceptable limitations

Amy told me that she would like a program to combine maths revision and question tools with flashcards. To start with I looked at existing software that Amy and I had used before for this purpose, and we thought of things that could be improved about it these were; [1][2]

* Algorithm for choosing the next card was too bias towards cards rated one.
* Takes lots of time to create recourses, if you make your own.
* All the preset packs weren't useful.
* Brainscape didn't do any randomised questions.
* Extra factors to determine how well the user knew the cards contents, which could be turned off.

However Brainscape showed features that Amy wanted to see in my program, which were custom cards, user ratings, deck statistics, duplicating cards and decks, and editing at runtime. She also thought that deck sharing with other users would be a nice feature however wouldn't be necessary for her as a user. She also wanted to see some maths tools alongside the flash cards such as creating graphs.

I then came up with new ideas , that I hadn't seen elsewhere , this included talking to my sister about what she thought would help her revise well [3] . After creating a list of possible features I presented this back to Amy to see what she thought and to see what else she wanted. [4]

After this I had a list of new ideas;

* View past answers to questions.
* Reward system with badges and/or stars.
* Select multiple decks to use at once.

We also decided that revision games such as snap, other more general revision tools, and motivational messages/images would be useful to Amy, but aren't as necessary as the maths section.

## Data sources and destinations

* The User will provide the answers and inputs that drive the program forward.
* Data will be sent to a database to be saved.
* Data will be opened from a database and used to display information to the user.

## ****Data Volumes****

My system will have to process data for one user. The total amount processed will depend upon how many flash cards they make or generate while using my program. It will have to process the answer for each flash card when it’s being run.

An average flashcard stored, will contain.

* The answer as a string, of length 255
* Previous answers, of length 255. Up to 5 unique answers.
* Card Type: as a char.
* Question: string of length 255.
* Equation string with length 50.
* Current user rating: short integer.
* Last 3 times for completion. Integer
* Overall card rating. short integer
* Card ID: long integer.
* Pack ID: long integer.
* Favourite: Boolean

Overall size per card approximately 3690 bytes.

With an average revision deck containing 100-500 cards including generated questions, this means per deck around 400kB-1.5MB needs to be stored.

As Amy could make up to 3 decks (one for each of her subjects), this puts total storage needed for cards at around 1-4.5MB.

As the rest of my data being stored (deck ids, stats, trophies etc) isn’t repeated as much its size is insignificant in comparison.

These cards are likely to be seen about five times over the course of the revision period. This means that 5-23MB of data will be processed in two months. This works out at 0.1-0.4MB processed each day.

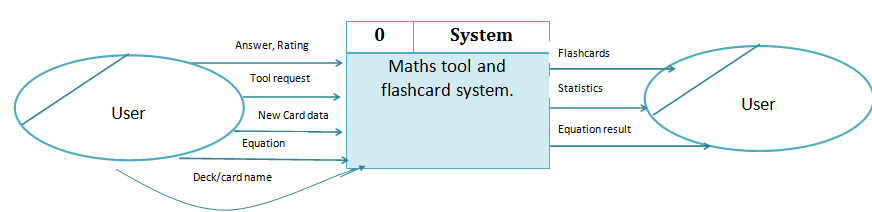
## Analysis Data dictionary

|  |  |  |
| --- | --- | --- |
| Data Name | Data Type | Example |
| Answer. | String | “x^2-2x+7” |
| Previous answers. | String | “x-2” |
| Card Type | Char | “G” |
| Question | String | “cos(2x)=” |
| Equation | String | “sin(x+2)” |
| Time | Real | 21 |
| Card Rating | Short Integer | 87 |
| User Rating | Short Integer | 3 |
| Card ID | Long integer | 12 |
| Deck ID | Long integer | 2 |
| Pack Size | Integer | 55 |
| Pack ID | Long integer | 22 |
| Pack name | String | “Identities” |
| Deck name | String | “A2 Maths” |
| Pack completion | Short Integer | 87 |
| Deck Knowledge | Short Integer | 45 |
| Deck completion | Short Integer | 55 |
| Pack Knowledge | Short Integer | 7 |
| Cardsseen | Integer | 567 |
| Achievement progress | Integer | 55 |
| Correct streak | Integer | 12 |
| User knowledge | Short Integer | 10 |
| Users name | String | 'Amy' |
| User completion | Short Integer | 22 |
| Favourite | Boolean | True |

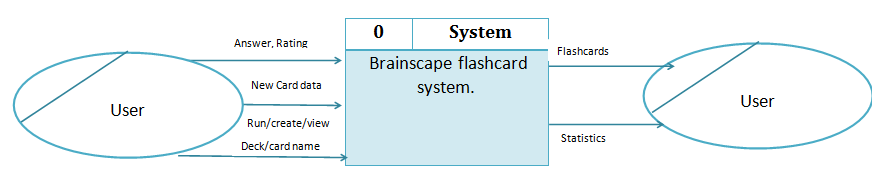
The knowledge sections are based on the card ratings, the completion is based off amount of cards unseen.

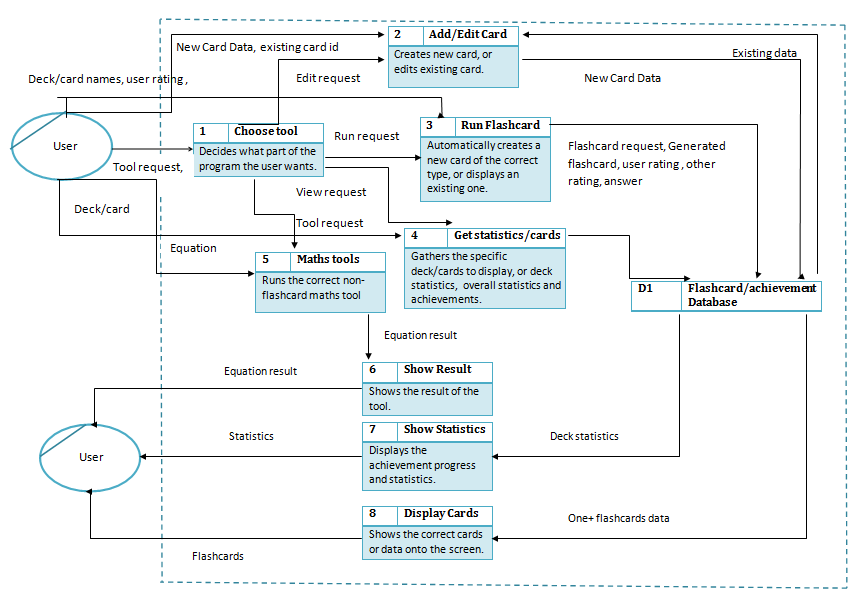
## Logical DFDs

Level 0- My system



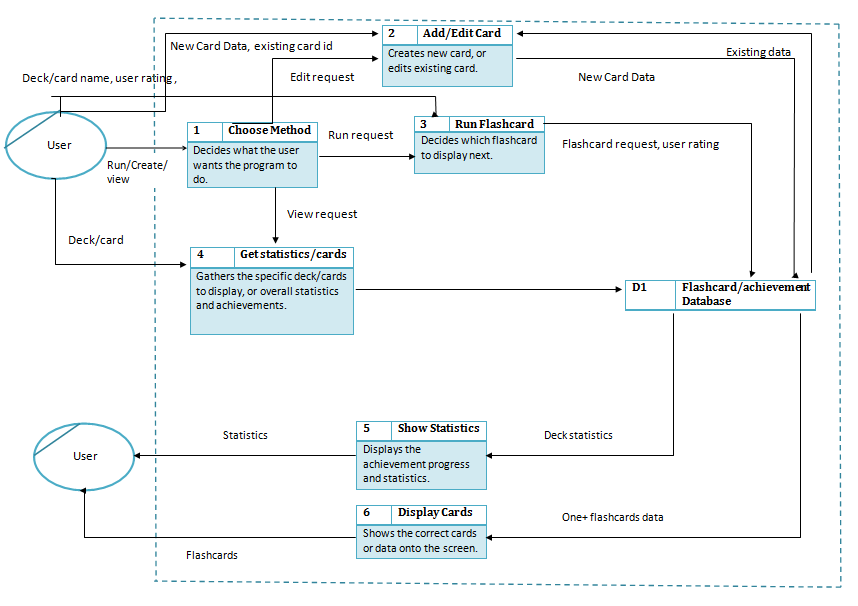
Level 0- Brianscape



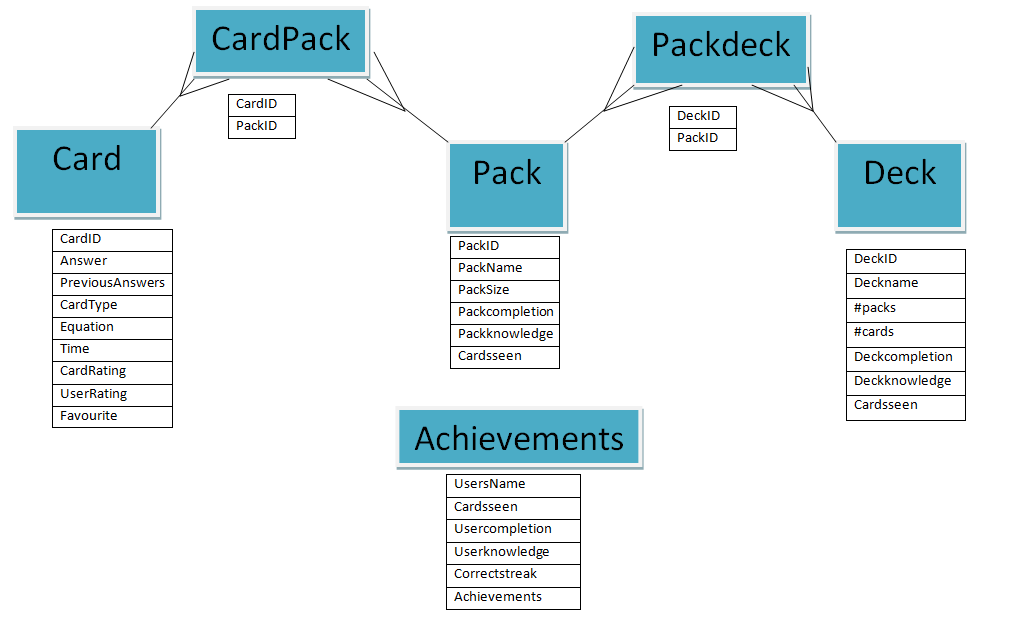


Level 1- My System

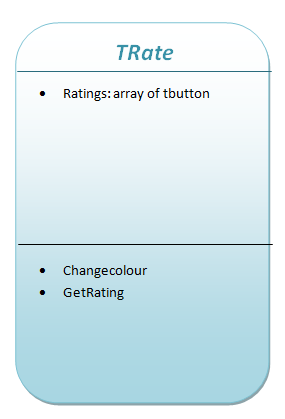
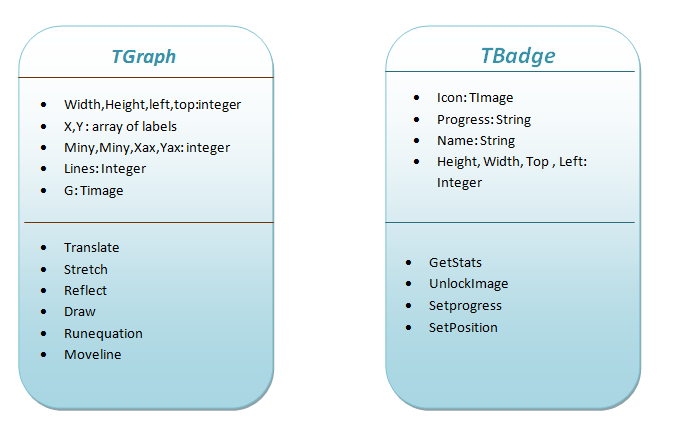
Level 1- Brainscape

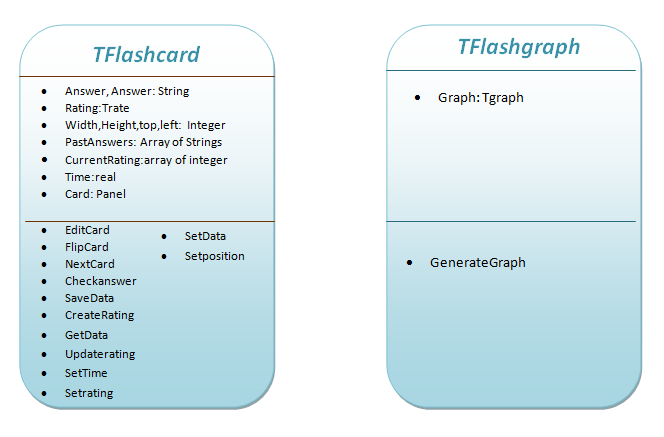


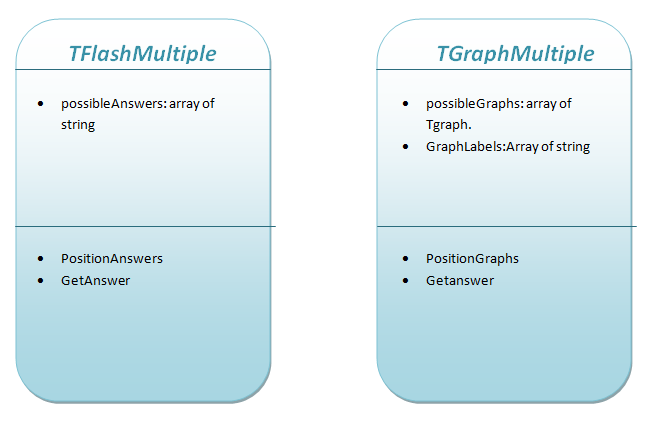
## E-R model

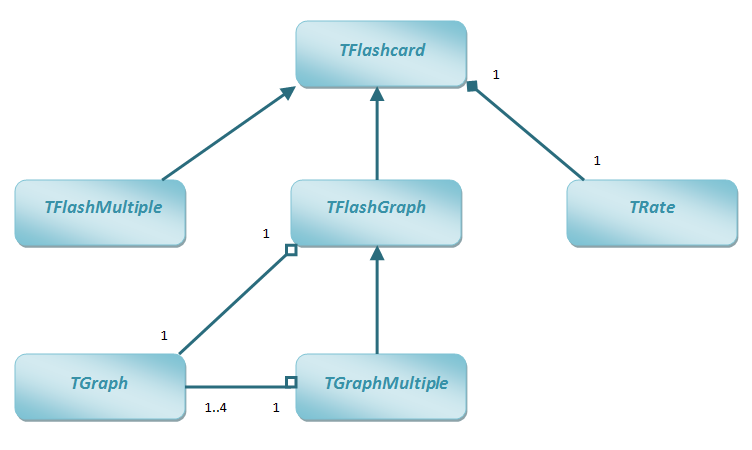


## ****Identification of Objects and Object analysis diagrams****

The flashcards amount other items will be an object in my program. As I'm having multiple kinds of flashcard i will have different subclasses for each of them in order to improve efficiency. 







## Objectives for the proposed system

Must

1. The Main focus of the program will be the A2 maths course, anything prebuilt must be based upon this.
2. The system must store flashcard data.
   1. This data must be stored in a user friendly way, that allows easy access to the flashcards while using a deck or editing one.
      1. Navigating around the program must be easy and intuitive
      2. Selecting, deleting and adding decks must be easy.
      3. Adding cards must be through a easy to use menu.
   2. The data must be editable from the application.
   3. The data store can't require additional paid software.
3. The system must allow for different packs within each deck.
   1. There must be some prebuilt packs in the program.
   2. Allow for user made packs.
      1. The cards can be of different formats.
   3. Have some topics which produce a randomised new question when the pack is run, which will give the user extensive practice at a certain topic.
4. Have an algorithm to determine which card gets shown next.
   1. This should be based on multiple factors ; user rating, time, and if the user was correct.
   2. Works with the randomised decks to allow for pre-generated cards based upon their previous ratings, and for difficulty of generated cards to increase as they improve.
5. Be able to toggle entering an answer , and other features if they are suitable.
6. The flashcards must be easy to use ,edit and view.
   1. Be able to see past answers to flashcards.
7. Be able to select multiple packs at once.
8. Be able to copy cards or add them to a favourite list.
   1. Be able to view the favourite list.
9. Track deck, pack and overall statistics.
   1. Award achievements, and badges based upon progress.
10. Offer certain maths tools such as graphing to the user outside the flashcards.
    1. This should work for functions including; trig, logs, exponentials, polynomials, modulus , and functions applied to bracketed areas.
    2. Inputted as a nY+k=........ , and a graph is drawn from it.
    3. Add multiple lines to the same graph.
    4. Graphs should be possible to add to a flashcard.
11. Operate on a windows computer.

Could but not required

1. Allow for deck sharing, using a simple and easy method.
2. Create revision games that use the flashcards.
   1. Eg ; Snap, timed questions, dominos , and memory.

## Realistic appraisal of the feasibility of potential solutions

Data Storage

|  |  |
| --- | --- |
| Database | Text/binary Document |
| Easier to code as I can use SQL to access the data. | Harder to code as I would have to think of ways to store the data so that I can access and save the correct parts easily and quickly. |
| Easy to include all the entity relationships that would make managing the data work. | More difficult to include these relationships |
| I have to use the data types that my database software allows me to. | I would be able to create my own data types to store which could make it easier to store what I want. |
| Much easier to achieve within my timescale. |  |

|  |  |
| --- | --- |
| Server-side DBMS | Desktop DBMS |
| Requires user to install new software which means installation would be harder and take longer and could require buying software. | Requires no additional software for the user, making installation and distributing my program easier. |
| Would be more expensive as I would also have to get a new software to use, as well as my user needing to. | Works well for a small number of users. |
| Works well with multi user databases. |  |

Program base

|  |  |
| --- | --- |
| Delphi - Programming tool | Microsoft Access - database package |
| I already know how to use this so wouldn't have to learn anything new. | I would have to learn how to program within a database software |
| I would be able to achieve everything on my objectives list | You can do less with it, so may not be able to do certain objectives. |
| Provides the visual aspect that I need to make my program user friendly. |  |

Platform

|  |  |
| --- | --- |
| Web-Based | Desktop based |
| Would be accessible from multiple devices | Only available on the installed computer without transferring files. |
| Harder to do as I'd have to consider multiple screen sizes, browsers , and data storage. | I know more about this so would be able to make my program. |
| Would require a server for the data to be shared across devices, which I don have access to. | Doesn't require any additional hardware for the program to work. |
| Hard to achieve as my database would have to allow for multiple devices to access and edit. |  |

Flashcards and visual items

|  |  |
| --- | --- |
| Object Oriented | Procedure oriented |
| Harder to initially program, but easier to reuse later on. | Easy to make work for single components but harder to reuse for multiple, or repeated use. |

## Justification of chosen solution

Database

I will be using a desktop DBMS; Microsoft access for my data store. I've chosen this as it doesn't require any additional software to be bought by my user. This means I can meet one of my objectives, and make my program easier to install. As it means I can distribute the required database without any additional setup being required or he user knowing how databases' work.

Using a database instead of a text/binary document to store my data means that I can easily use dynamic SQL to retrieve and store the data that I need at that moment, as the data I require will change thought running my program. Also databases provide me the best way of ensuring by entities are properly related to each other so I can have the right structure for my program. I will however have to make sure that all the data I want storing can be done using the types that access providers.

Program base

I've chosen to use Delphi as my base for creating the program over using a database package such as access directly. This is due knowing more about programming in a programming tool so would be unable to complete the task using a solely database software. Also my program requires more advanced operations to happen that access couldn't provide. Such as a visual representation of the flashcards, graphs, statistics and changing menus.

Platform

I will be making my program work on a windows computer. This is due to my user not requiring it to work for other operating systems or mobile devices. I also don't have the knowledge required to make my program work on mobile devices with all the factors that it needs so can't provide this as an extra feature.

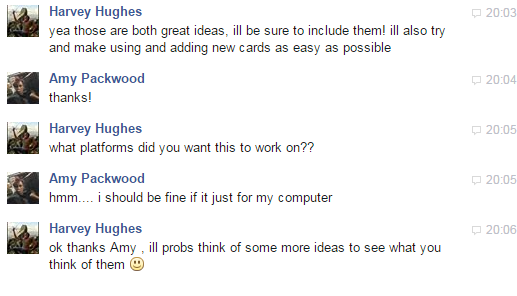
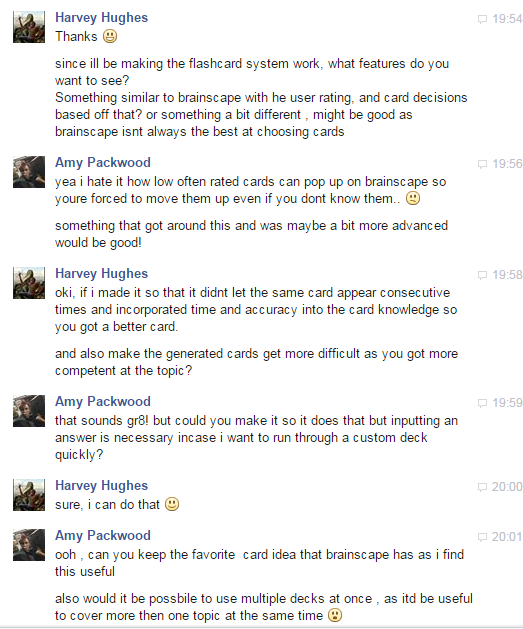
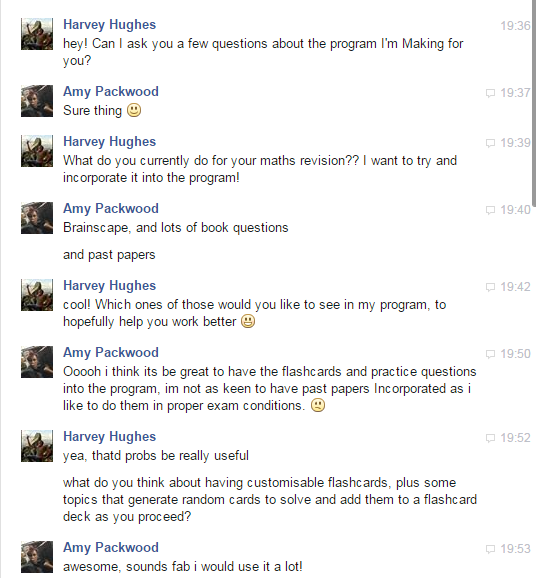
My program will be desktop based as for my user I only need it to work on her windows computer. Also if I were to use a web based program I would have to learn how to write it in a different language, using a different programming tool and a better multi user database system. This means it would be far too complicated for me to achieve a web based solution to this problem.

Objects

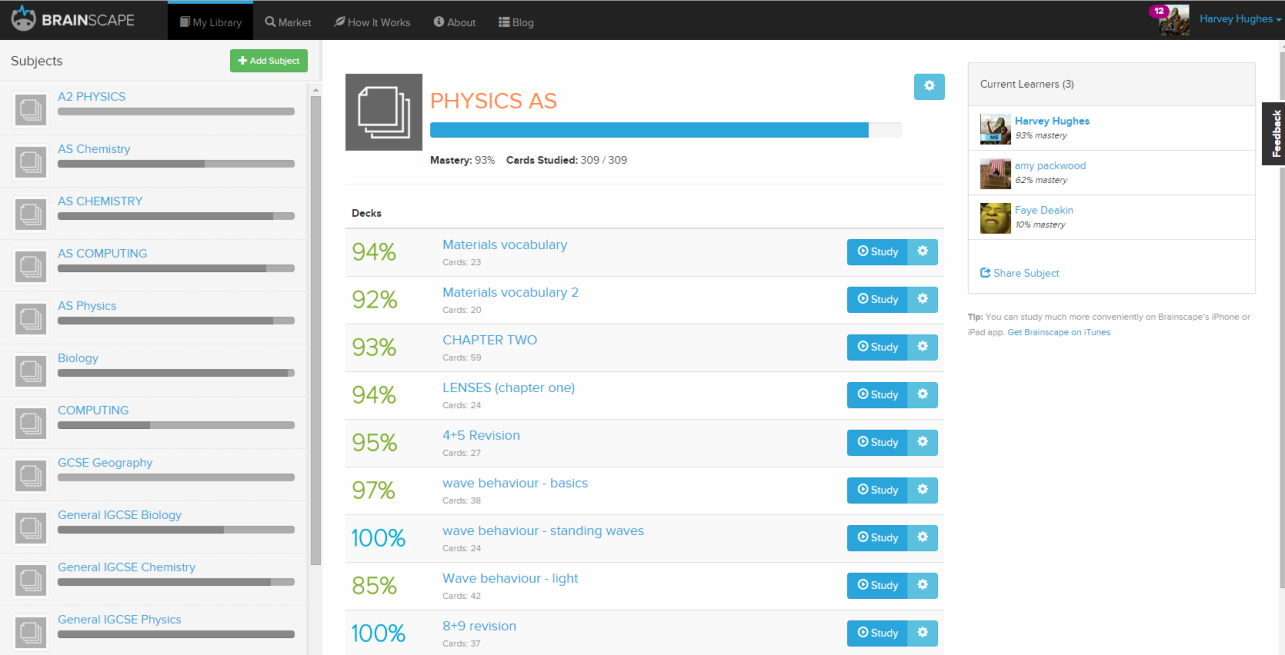
My program will feature many of its components as an object opposed to using lots of procedures with local variables. This is due to having to dynamically create lots of these as each time a new deck is created, or they open a section from the menu a new object will have to be created. Having these as objects means that I can easily organise the creation, destruction, methods and attributes of them making sure that each new object functions properly and is easy to dynamically create

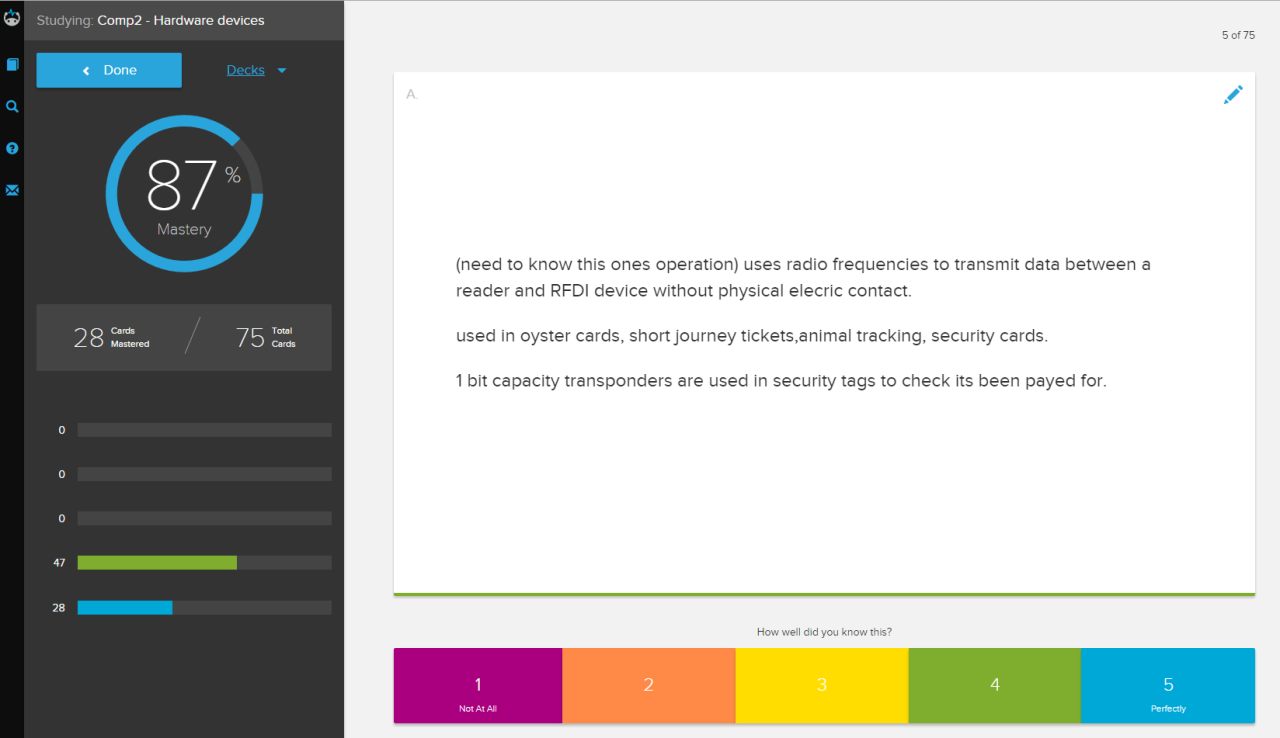
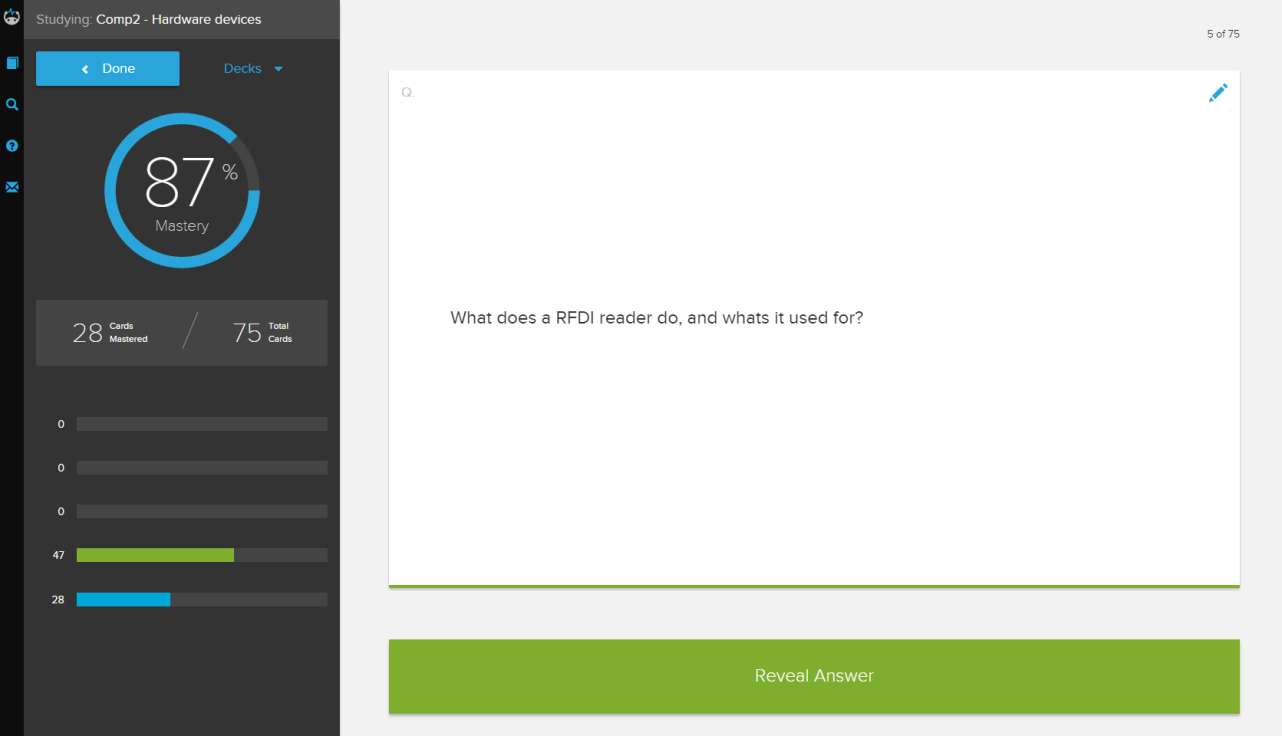
## Appendix

[1] Conversation I had with Amy having already decided what the program would be based on.



[2] Pictures I took from Brianscape.





[3] Transcript from conversation with my sister.

Harvey: Jasmine what do you think should be improved with brainscape other then these things that­­­­­­ Ive already thought of.

Jasmine: Ah i like the ideas youve got there, especially the new way of choosing a card. I think making flashcards into a game would be fun if thats possible?

Harvey:What kind of game do you mean?

Jasmine: Something like dominos, or memory or you could even do snap!

Harvey: Yea that'd be a pretty good idea and probs make you more motivated and it should be possible for me to do thanks.

Jasmine: You havent got anythign about deck sharing, i found that pretty useful.

Harvey: I didnt think about that since it might not be as useful for my program, but I'll see what Amy thinks of the idea.

Was there anythign else?

Jasmine: i cant think of anything that would make the generated questions and flashcard better but have you thought about having any other revision tool?

Harvey: I thought i could have a timetable of exam dates or something similar, but think the rest of the program would be pretty useful.

Jasmine: Having a timetable for revision or exam times and maybe a to-do list I think would help as it puts more revision tools in one place. Thats all my ideas though.

Harvey: Okay thanks! I'll ask Amy what she thinks of them.

[4] Presenting Amy with the new ideas from me and my sister.

