

Applying investigational skills

Consider the following scenario:

A school has decided to introduce a smart card system for its pupils to pay for food and drinks in the school's dining hall. It has asked a programmer to design the system for the school.

Using **decomposition**, the whole problem is the introduction of the smart card system, but the programmer must break the problem down into manageable chunks. The system must be able to:

- allow parents to add money to their child's smart card
- allow pupils to pay for their food and drink
- update the amount of money remaining on the card.

So now the programmer has three sub-problems, but each of the three is quite complex and can be broken down further.

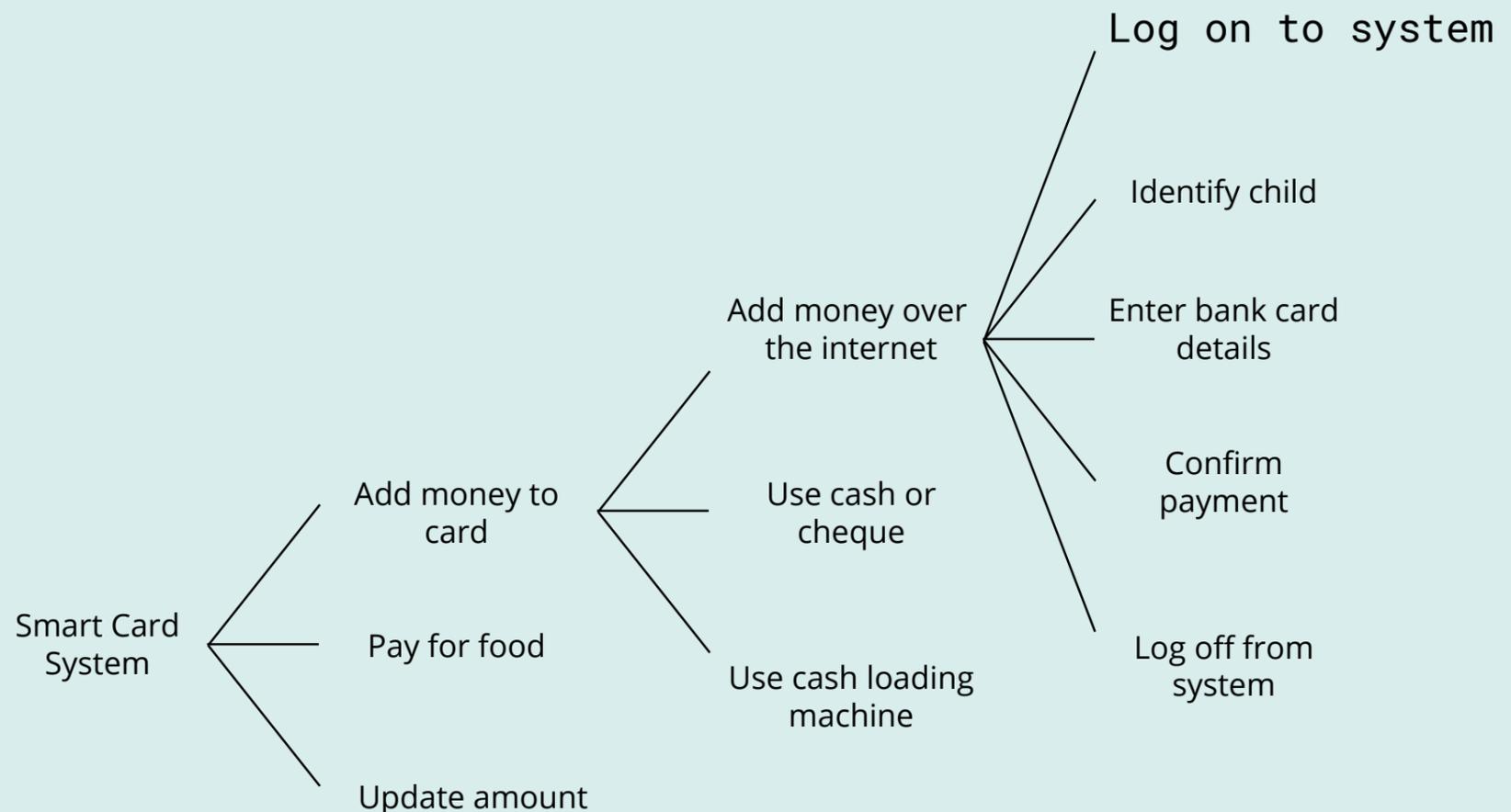
'Allow parents to add money to their child's smart card' can be broken down into the following problems:

- add money over the internet
- add money by giving cash or a cheque to the pupil, to take to the school office
- add money by allowing the pupil to use a cash-loading machine in the dining hall.

'Add money over the internet' can be broken down into the following problems:

- log on to the school's payment system
- identify the child who uses the smart card
- enter the details of the bank card to be used for payment
- confirm the payment
- log off from the school's payment system.

This process has broken part of the problem into distinct sub-problems that can be solved and combined to form part of the solution to the original problem.



Looking at the tasks that need to be carried out at this level, the logon and logoff facilities will be required in many other functions of the solution – this is an instance of spotting a pattern. The authentication routines will be common modules that will be called throughout the program.