



12 Pages

1. Please give the information requested in the boxes below.
2. Enter your name in the space provided to the right, fold the paper over and seal down (using the label provided) to hide your name.
3. The answers must be legibly written. The Examiners can either read your script or have it typed and the expense of typing will be charged to the candidate.
4. Write your answers in ink.
5. Begin each answer on a fresh page. Do all rough work in the answer book and cross it through.
6. No part of this book is to be torn out; it must be handed in complete.
7. If more books than one are used, fasten them together using the tags available from the invigilator(s).
8. **Every book must be handed in. In no circumstances may answer books, used or unused, be taken out of the examination venue.**

1	Student Number	1141924
2	Title of Examination Paper	object oriented programming
3	Module Code	M2X8523
4	Date and Time of Examination	18/05/18 9am
5	Examination Venue	Townhill campus Gym room
6	Seat Number	6016
7	Number of Answer Books handed in	1

[illegible]

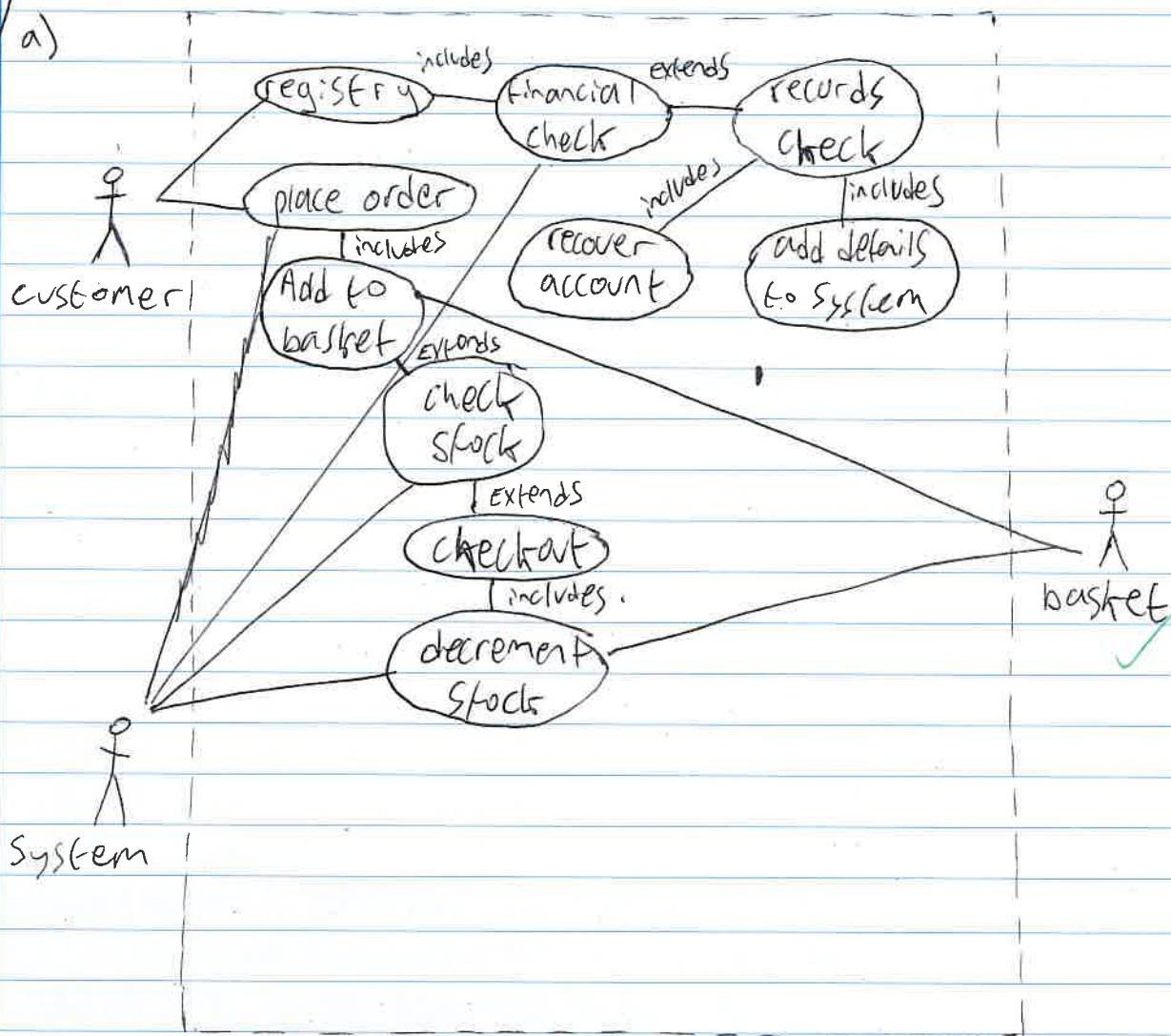
Q3/ ~~AGILE methodologies, for example~~

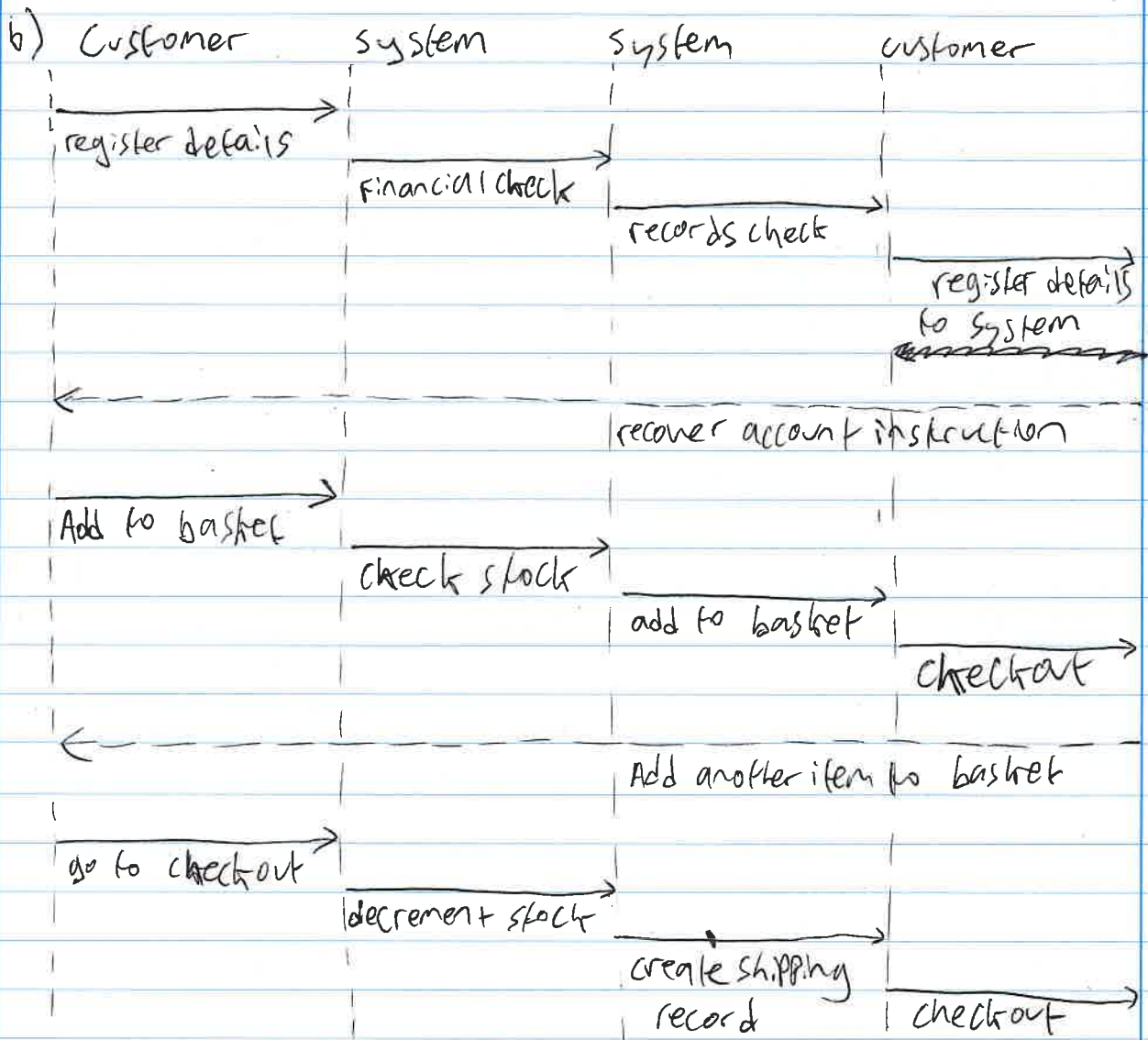
the AGILE manifesto defines four key aspects of AGILE methodologies, inception, elaboration, development and . The Agile workflow begins with establishing a set of requirements which are analysed and developed into a design. Upon completion the design is implemented and tested through its product life cycle.

Agile approaches project development with a more fluid and adaptive approach (than the predictable unified process). Agile focuses less on documentation and whilst using some aspects of unified modelling languages UML is more often associated with UML.

Q3-5

Q4/a)





Q4-11



Q1 //controlSystem.h

```
#include <stdlib.h>
#include <stdio.h>
#include "customer.h"
#include "order.h"
```

class controlSystem()

{ private:

public:

int customerID, itemCode, quantity, orderID;

int day, month, year;

bool shipStatus;

int getCustomerID(int orderCustomerID);

int getItemCode(int orderItemCode);

int getQuantity(int orderQuantity);

int getDay(int orderDay);

int getMonth(int orderMonth);

int getYear(int orderYear);

bool getStatus(bool orderStatus);

int ammendQuantity(int orderID, int quantity);

}

5 ✓

//controlSystem.cpp

controlSystem::~controlSystem()

{ }

~~controlSystem~~

controlSystem::getCustomerID(~~int orderItemCode~~) (int orderCustomerID)

{

customerID = orderCustomerID;

return customerID;

}

controlSystem::getItemCode(int orderItemCode)

{

itemCode = orderItemCode;

return itemCode;

}

2 ✓

```
ControlSystem:: getQuantity (int orderQuantity)
{
    quantity = orderQuantity;
    return quantity;
}
```

```
ControlSystem:: getDay (int orderDay)
{
    day = orderDay;
    return day;
}
```

```
ControlSystem:: getMonth (int orderMonth)
{
    month = orderMonth;
    return month;
}
```

```
ControlSystem:: getYear (int orderYear)
{
    year = orderYear;
    return year;
}
```

```
ControlSystem:: getStatus (bool orderStatus)
{
status shipStatus = orderStatus;
    return shipStatus;
}
```

```
ControlSystem:: amendQuantity (int orderID, int quantity)
{
cout cout << "new quantity" << endl;
    cin >> quantity;
    return quantity;
}
```