

## Analysis and design

### Contents

|   |    |
|---|----|
| Analysis and design .....   | 1  |
| background .....  | 3  |
| Problem Definition.....   | 5  |
| Basic overview .....  | 5  |
| General aims .....  | 5  |
| Student.....  | 5  |
| Staff – Subject teachers .....  | 5  |
| Staff - Form tutor .....  | 5  |
| Staff - Head of year .....  | 6  |
| Staff - Head teacher .....  | 6  |
| Staff – Admin.....  | 6  |
| Staff – Other.....  | 6  |
| Data .....  | 6  |
| Limitations.....  | 7  |
| Objectives.....   | 8  |
| General Objectives.....   | 8  |
| Student objectives .....  | 8  |
| Staff objectives.....   | 8  |
| Admin objectives.....   | 9  |
| Justification of the proposed solution .....  | 10 |
| Solution 1 – paper based/manual.....  | 10 |
| Solution 2 – buy existing system ( <a href="http://www.schoolrewards.co.uk/">http://www.schoolrewards.co.uk/</a> )..... | 10 |
| Solution 3 – application based (access) .....   | 11 |
| Solution 4 – application based (excel) .....  | 11 |
| Solution 5 – Visual Basic .....   | 11 |
| The decision .....  | 12 |
| Data structures and methods of access.....  | 13 |
| Access.....   | 13 |
| File structure .....  | 13 |
| User Interface     5.....   | 16 |
| Student form .....  | 16 |

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

|  |    |
|--|----|
| Teacher form.....                          | 18 |
| Student information Form .....             | 19 |
| Admin Form .....                           | 19 |
| Admin – Staff form.....                    | 20 |
| Admin – Students form .....                | 21 |
| Admin – rewards form .....                 | 21 |
| Hardware and Software requirements 2 ..... | 23 |
| Processing stages 10 .....                 | 24 |
| Importing files .....                      | 24 |
| Logging on .....                           | 24 |
| Changing passwords .....                   | 25 |
| Adding a student/staff member/reward ..... | 25 |
| Points procedure.....                      | 26 |
| Buying a reward .....                      | 26 |
| Searching.....                             | 27 |
| Evaluation criteria 3 .....                | 28 |

## ***background***

### GUIDANCE FOR CANDIDATES

You are expected to:

1. Read and analyse the given scenario and develop a solution which may be enhanced by making any assumptions you see fit. Any assumptions you make should be clearly explained and justified in your documentation.
2. Produce a solution which **MUST** include original coding. Solutions which do not include original coding will not be acceptable.
3. Produce a fully documented solution to the given problem in accordance with the requirements listed in CG2 (pages 17-19 and pages 41-51 of the specification). Credit will be given for the quality of your solution.

JD\*(A11-1102-01)

2

(1102-01)

Greenparks School Reward Scheme

Many schools have introduced reward schemes as a strategy to motivate their pupils and to encourage good behaviour. The headteacher of Greenparks School has decided to set up such a

scheme that she hopes will encourage the pupils to improve their attendance and behaviour and

lead to an overall improvement in the school's results.

For example, pupils will be able to earn reward points for:

- Good attendance at school
- Good behaviour in lessons
- Working hard in class
- Gaining good marks in tests and examinations
- Taking part in sporting and musical activities
- Representing the school in competitions and events
- Taking part in after school clubs and activities
- Taking part in activities such as litter patrols and community based projects.

Points may be deducted for poor behaviour.

Points will be added to or deducted from a pupil's account by a form tutor, subject teacher or

head of year.

Pupils will be able to view their own account.

The Greenparks Reward Scheme will work by allowing the pupils to redeem their points for rewards that can be chosen from the given options.

The headteacher has commissioned you to create a computer based system which will:

- Allow users to log in either as staff or pupil
- Allow staff to add points to or deduct points from a pupil's account
- Display a pupil's points balance
- Display the reward or rewards available for a pupil's current points total
- Update a pupil's points total when rewards are redeemed.

3

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

| Reward         | Points needed |
|----------------|---------------|
| Set of pens    | 25 points     |
| CD voucher     | 60 points     |
| Memory stick   | 100 points    |
| Ice Skating    | 110 points    |
| Computer Game  | 150 points    |
| Driving lesson | 250 points    |
| Digital camera | 300 points    |
| MP3 player     | 400 points    |
| Games console  | 500 points    |

## ***Problem Definition***

### **Basic overview**

The project is to create a reward scheme that can be used by green parks school. The idea is for it to be adaptable so students and staff can be added, rewards changed and for staff to be upgraded to admin. It will be controlled by a couple admin who may or may not be teachers. For it to be accessible by everyone, so that students can check their own scores without an intermediary figure. This being said the system is to be only accessible while in school and only teachers and admin will be able to see all students' files instead of just their own. There is to be a point system in which the points can be redeemed for prizes by students. The system is also to be secure so that it cannot be changed by an unauthorised user.

### **General aims**

- Users must- be able to log on using there user ID and password but only while logged onto their network area. If there ID does not match their collage logon then does not login
- To be professional
- For it to be user friendly
- For it to be idiot proof
- For it to take up small amounts of memory
- For it to load quickly
- Have timed log off
- Strong security
- To be able adapt

### **Student**

Must be able to:

Check the amount of points that they have by logging on  
See the reasons for the point increases and decreases  
Redeem points on the programme  
When redeeming their points they are asked again for their password

### **Staff – Subject teachers**

Must be able to:

Add points  
Deduct points  
See the points of students  
Have a limited amount of points

### **Staff - Form tutor**

Must be able to:

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

- Add points
- Deduct points
- See the points of students
- Have a limited amount of points

Could be able to do:

- Print a document with there whole forms points in 1 report
- Print out documents for each student to be sent home to parents

### **Staff - Head of year**

Must be able to:

- Add points
- Deduct points
- See the points of students
- Have a limited amount of points

Could be able to:

- Receive reports of the students who are performing best and worst
- Create reports containing statistics in relation to the points of the students in their year
- Create reports for each of the students at the end of each year for the schools hardcopy permanent reports

### **Staff - Head teacher**

Must be able to:

- Add points
- Deduct points
- See the points of students

Could be able to:

- Create reports containing statistics in relation to the points of the students
- Get reports of any teachers distributing merits at an un sustainable rate

### **Staff – Admin**

Must be able to:

- See the points of students
- Change prizes
- Change cost of prizes
- Add and edit students
- Give staff more points to distribute

### **Staff – Other**

Could be able to:

- Deduct points for bad behaviour

### **Data**

Must be encrypted so that it cannot be accessed by unauthorised personnel  
Must be editable

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

Could be easily transferable from the form onto documents for printing

### **Limitations**

No access to parents or rest of general public  
Students can only see their own records  
Students cannot edit any records or add or subtract points  
Staff will have an allotted amount of points to distribute  
No access from anywhere outside of school  
Cannot Internet based because that would make it vulnerable  
Only admin will be able to alter the system  
When points are deducted they are not transferred to said teacher  
Cannot cost more than £100  
When redeeming points for prizes the student must not be able to go into negative points  
Teachers will not be able to take students into negative points  
Only admin can designate more points to a teacher  
Teachers will not be able to transfer points between them selves  
Only teachers can be made admin  
Cannot be over 20 MB  
Only passwords will be encrypted  
Will not be backed up  
Will not involve a mechanism for delivering prizes  
Will not rebuild files  
Will not check which computer is being used to logon  
Will not have the capacity for new fields to be added  
Will not have the capacity to go over 2000 students  
Will not have the capacity to go over 150 teachers  
Will not have the capacity to go over 20 different rewards  
Will not work on Apple Mac's

## ***Objectives***

### **General Objectives**

1. To make it professional by
  - a. using a house style
  - b. keeping colours to a minimum
2. To make it user friendly by
  - a. using clear and understandable language
  - b. Making everything accessible by hot keys or left clicking
3. To make it load quickly by
  - a. keeping load time under 10 seconds
4. To keep it size small by
  - a. keeping its size under 20 MB
5. To make it fool proof by
  - a. data validating all input and putting in fail safes so they can't break it
6. To have strong security
  - a. timed log off
  - b. password protect
  - c. must have passwords encrypted
  - d. make sure the passwords are alpha-numeric
7. To be able to adapt
  - a. must be able to add and remove all types of user
  - b. must be able to change the amount of points for teachers
  - c. must be able to change prizes and their price
  - d. edit the records of existing users
  - e. Must be able to upgrade staff to admin
8. To not go over the budget of £100

### **Student objectives**

1. Must be able to see his/her points
2. Must be able to see why he/she had points deducted/added
3. Must be able to change his/her password
4. Must be able to redeem points for rewards

### **Staff objectives**

1. Must be able to see their remaining points



Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

2. Must be able to see the points of all students
3. Must be able to change their passwords
4. Must be able to give and take away points

### **Admin objectives**

1. Must be able to add and edit staff records
2. Must be able to add and edit student records
3. Must be able to add and edit rewards
4. Must be able to upgrade a member of staff to admin
5. Must be able to see decrypted passwords so that they can reset them
6. Must be able to give staff more points

## ***Justification of the proposed solution***

### **Solution 1 – paper based/manual**

#### **Pros**

It would be cheap to set up  
It would be hard to change the scores  
Simple to remove and add students  
Would not need to worry about making the system look professional aesthetically as it would not have a user interface  
Would not take up a lot of room on the network as it would not have a digital presence anywhere  
Wouldn't have to worry about passwords as the students would either just have to turn up and have their face matched to that of their file or show their student ID if they have one  
Prizes and their price could be easily changed as they would be just a poster or sign

#### **Cons**

It would take a lot of time to manage and file correctly  
It would be hard for students to check up on their points and would take a long time  
Points would have to be transferred from each member of staff at certain times and then added to central filing and points could be lost  
Could data could be easily removed  
Students could not redeem their points at a moment's notice and would have to file a claim  
If a student got into the records they could easily see the file of anyone

### **Solution 2 – buy existing system (<http://www.schoolrewards.co.uk/>)**

#### **Pros**

Quick to get running  
Simple procedures for adding points  
Quick times for transferring points, though program speed unknown  
Looks professional  
Is apparently secure requiring a teacher's card and code to operate points transfers

#### **Cons**

Requires more than 20 MB  
Is over budget  
Requires cards which could be misplaced

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

### **Solution 3 – application based (access)**

#### **Pros**

Would be quite simple to set up as it comes with many templates that would only have to be slightly altered to make them in to what I want  
Adding students would be easy as it is part of the programming  
The admin would probably have prior experience with the program and therefor wouldn't have much trouble with the program  
Software is able to data validate already

#### **Cons**

Would have a slow loading time, greater than what I want  
Prone to crashing if too many people on at one time meaning that it would break often and would necessitate a queue to check the system  
Would take up significantly more space then I would like  
Can't protect different parts of it separately so if they can see some they can see all and therefore edit all

### **Solution 4 – application based (excel)**

#### **Pros**

Would be simple to set up  
Admin would probably have prior experience with the program  
All the different sheets could be encrypted separately  
Opens quickly  
Easily adaptable, can add students simply

#### **Cons**

Would require a large number of spread sheets and therefore would take up too much memory  
Students would be able to alter the sheets they could see  
Not a very professional astatically

### **Solution 5 – Visual Basic**

#### **Pros**

Easy to make professional as colour is completely changeable and the different components all look professional.  
Can set many different types of triggers for different actions  
Can easily code the different inputs to validate inputted data  
Opens quickly  
Takes up little space

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

Can be made to add and remove students, rewards and staff  
Can easily alter data  
Can be made to have encryptions on each separate parts of the programme making them password protected  
Can be set to verify if the user is on his/her network area  
Can be set to ask for password a second time when redeeming points  
Free

## **Cons**

Will take the longest to create

## **The decision**

I have decided to go with solution 5 as it ticks the most boxes whilst having the least cons, the only one which is time is of no consequence as it should still be deliverable within the deadline. Solution 1 the paper-based system is the second as it would be the simplest to run and the security concerns it raises aren't severe enough it also passes all of the most important objectives without including any of the limitations. Next come solutions 3 and 4 which are equally poorly suited for their limitations on simultaneous use, for their size and for in the case of 3 the fact that you can't encrypt it in sections. Last is solution 2 for though it does a lot of the pros very well it is failed completely by the fact that it is over budget. The main plus for solution 5 is that it can be used to create data grids for the display of information, which can be interacted with, it can create labels, text boxes, buttons, functions and procedures that can have a wide range of different triggers. It can also make text boxes and labels invisible till needed and make forms unable to be bypassed.

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

## ***Data structures and methods of access***

### **Students**

(studID, Surname, forename, DOB, Points, Password, form)

Maximum of 2000 records

Size per file 93 bytes

Total size 186KB

### **Staff**

(staffID, surname, forename, points left, password, form)

Maximum of 150

Size per file 82 bytes

Total size 164KB

### **Rewards**

(rewardID, name, cost)

Maximum of 20

Size per file 63 bytes

Total size 126KB

### **Points redeeming**

(studID, rewardID, points, cost)

### **Points giving**

(staffID, StudID, points left, points, points exchanged, reason)

## **Access**

Access will be done by using direct access to find the data as I believe that it will be simpler and easier than using a sequential system. The reason for this is that if I were using a sequential system I would need to keep records ordered which would make adding data more complicated but with a serial one the new records could just be added to the bottom of the file.

## **File structure**

### **Students**

| Field name  | Data type  | Length (bytes) | Format     | Validation                           | Purpose                              | Comments         | Example data |
|-------------|------------|----------------|------------|--------------------------------------|--------------------------------------|------------------|--------------|
| Student NO. | short      | 2              | 0000       | Unique number automatically assigned | Uniquely identifies a student record | Storage position | 0012         |
| Student ID  | Integer    | 4              | 000000     | Unique number automatically assigned | Uniquely identifies a student record | Logon ID         | 112544       |
| Forename    | String(20) | 40             | LLLLLLLLLL | Must be letters                      | Other way of                         |                  | Simon        |

Title: CG2-Green park school reward scheme  
 Centre: Shrewsbury sixth form college (29285)  
 Candidate: Simon Bellows (1054)

|          |            |    |                  |   |   |  |           |
|----------|------------|----|------------------|---|---|--|-----------|
|          |            |    |                  |   | finding student if name unknown                 |  |           |
| Surname  | String(20) | 40 | LLLLLLLLLLLLLLLL | " "   | " "   |  | Bellows   |
| DOB      | Date       | 8  | dd/mm/yyyy       |   | Gives students date of birth                    |  | 18/05/95  |
| Points   | short      | 2  | 0000             |   | Stores the students number of points            |  | 0134      |
| Password | String(10) | 20 | LLLLLLLLLL       | Must have letters and numbers and be at least 6 characters long | Is what they need to input to enter the program |  | Password1 |
| Form     | String(4)  | 8  | LLLO             |   | Is the students form                            |  | SEH1      |
| Year     | byte       | 1  | 00               | Must be between 7 and 11  |   |  |           |

## Staff

| Field name  | Data type  | Length (bytes) | format           | validation  | purpose  | comments | example   |
|-------------|------------|----------------|------------------|---|--|----------|-----------|
| StaffID     | Short      | 2              | 0000             | Must be a 4 character number                                    | Unique identifier for the member of staff                  |          | 1164      |
| Forename    | String(10) | 20             | LLLLLLLLLL       | Must be letters   | To make it easier to find the member of staff              |          | Jane      |
| Surname     | String(15) | 30             | LLLLLLLLLLLLLLLL | " "   | " "  |          | Appleton  |
| Points left | Short      | 2              | 0000             | Must be a number  | To track the amount of points they spend                   |          | 0478      |
| Password    | String(10) | 20             | LLLLLLLLLL       | Must have letters and numbers and be at least 6 characters long | To stop others from getting onto their area                |          | Password1 |
| Form        | string(4)  | 8              | LLLO             |   | To show what form they are in                              |          | JAP1      |
| Admin       | boolean    | 2              | True or false    | Must be a true or false   | To decide whether they are able to get onto the admin form |          | true      |

## rewards

| Field name | Data type  | Length (bytes) | Format                                       | Validation                           | Purpose   | Comments | example            |
|------------|------------|----------------|--|--------------------------------------|---|----------|--------------------|
| RewardID   | byte       | 1              | 000  | Unique number automatically assigned | A unique identifier for the reward              |          | 167                |
| name       | string(30) | 60             | LLLLLLLLLLLLLLLL<br>LLLLLLLLLLLLLLLL<br>LLLL | Must not be empty                    | To tell the student what the reward is          |          | A big box 'o' pens |
| cost       | short      | 2              | 0000   | Must be a number                     | The number of points required to buy the reward |          | 0025               |

Title: CG2-Green park school reward scheme  
 Centre: Shrewsbury sixth form college (29285)  
 Candidate: Simon Bellows (1054)

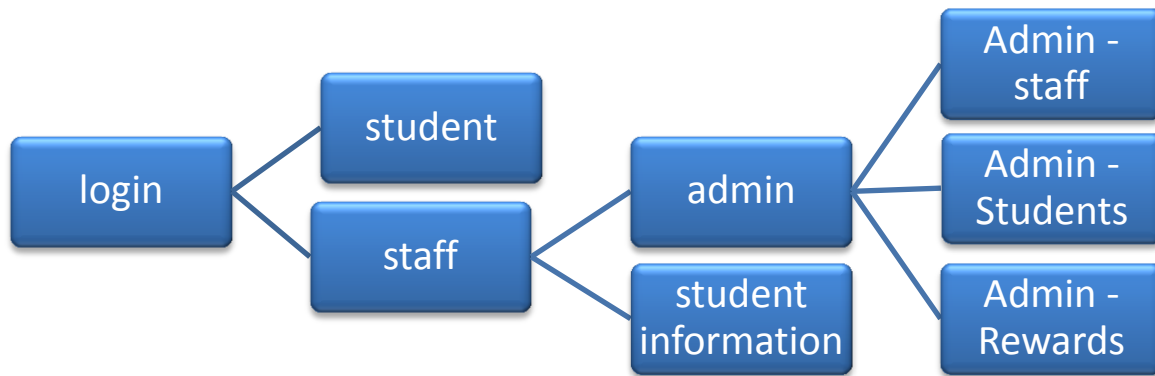
### Points procedure

| Field name              | Data type  | Length (bytes) | Format   | Validation        | Purpose                                       | Comments | example             |
|-------------------------|------------|----------------|--|-------------------|---|----------|---------------------|
| Stud NO.                | Short      | 2              | 0000   | Must be numeric   | Find the students data                        |          | 0012                |
| Staff NO.               | Short      | 2              | 0000   | Must be numeric   | To find staff's data                          |          | 0143                |
| Points left             | Short      | 2              | 0000   | Must be numeric   | To check if staff has enough points           |          | 0347                |
| Points                  | Short      | 2              | 0000   | Must be numeric   | To add points to                              |          | 0056                |
| Points awarded/deducted | Short      | 2              | 0000   | Must be numeric   | To determine how much points were transferred |          | 0001                |
| reason                  | string(25) | 50             | LLLLLLLLLL<br>LLLLLLLLLLLL<br>LLLLLLLLLLLL<br>LLLLLLLLLLLL<br>LLLLLLLLLLLL | Must not be empty | To record the reason for the points           |          | He stabbed a kitten |

### Rewards procedure

| Field name | Data type | Length (bytes) | Format | Validation      | Purpose                             | Comments | example |
|------------|-----------|----------------|--------|-----------------|-------------------------------------|----------|---------|
| Stud NO.   | Short     | 2              | 0000   | Must be numeric | Find the students data              |          | 0012    |
| RewardID   | Short     | 2              | 0000   | Must be numeric | To find rewards data                |          | 0143    |
| Cost       | Short     | 2              | 0000   | Must be numeric | To check if staff has enough points |          | 0347    |
| Points     | Short     | 2              | 0000   | Must be numeric | To add points to                    |          | 0056    |

## User Interface 5



### Logon form

|  |   |
|--|---|
|  | <p>picLogo</p> <p>txtUserName</p> <p>btnLogin</p> <p>btnExit</p> <p>txtPassword</p> |
|--|---|

txtUserName – must be entered

txtPassword – must be entered

btnLogin – checks both text boxes are not empty, if so report error and request input if user name numeric then process as student otherwise process as staff open student/ staff file as appropriate and find user

if record not found report error and request re-input check passwords match

if not equal then report error and request re-input

save record in memory and open staff or student form as appropriate

btnExit – closes the form

### Student form

|  |  |
|--|--|
|  | <p>picPhoto</p> <p>lblName</p> <p>grdRewards</p> <p>grdRecords</p> |
|--|--|



Title: CG2-Green park school reward scheme  
 Centre: Shrewsbury sixth form college (29285)  
 Candidate: Simon Bellows (1054)

picPhoto – contains the school photo of the student

lblName – contains the students name and points

upon the loading of the form or the buying of a reward the label will be refreshed and the points of the student and their name will be put in the label

dgdRewards – data grid containing the different rewards and their prices. When the student clicks on a rewards to buy it they must re-enter their password. When a cell is clicked on the information for the corresponding reward will be imported. If the student has enough points a messagebox will open asking them are they sure they want to buy it with Yes/No buttons. If yes they will buy the reward

dgdRecords – a data grid containing their past point transactions from teachers showing the teacher the amount and the reason

btnPassword – when pressed makes grpPassword visible and changes the buttons text to execute, when pressed a second time if the textboxes contain the right information the password will be changed and grpPassword will disappear and the text of the button will revert to change password. Also a message box will pop up telling the student that the password has been changed

txtOldPassword – must be the students current password

txtNewPassword1 – must be at least 6 characters long and be alphanumeric

txtNewPassword2 – must be the same as txtNewPassword1

grpPassword – contains txtOldPassword , txtNewPassword1 and txtNewPassword2 and is invisible upon opening the document

btnExit – when pressed closes the form showing the login form again

## Teacher form

The screenshot shows a Windows-style application window titled "Teachers Form". It contains several sections and controls:

- Welcome Message:** "Welcome Joe Blogs you have 0000 points left".
- Transfer Points Section:** Includes a table with columns "StudID", "Points", and "Reason". Below it is a large greyed-out area.
- Students Section:** Includes a table with columns "StudID", "Forename", "Surname", and "Points". Below it is a large greyed-out area.
- Buttons:** "Execute points transfer", "change password", "Admin Form", and "Exit".
- Password Change Section:** Includes labels "change password", "old password", "new password", and "repeat new password", each followed by a text input field.

Arrows point from the following labels to specific UI elements in the form:

- picTeacher - points to a small rectangular area in the top left.
- lblName - points to the welcome message text.
- grdTransferPoints - points to the "Transfer Points" table.
- grdStudents - points to the "Students" table.
- btnPointsTranfer - points to the "Execute points transfer" button.
- btnPassword - points to the "change password" button.
- grpPassword - points to the password input fields.
- txtOldPassword - points to the "old password" text box.
- txtNewPassword1 - points to the "new password" text box.
- txtNewPassword2 - points to the "repeat new password" text box.
- btnAdmin - points to the "Admin Form" button.
- btnExit - points to the "Exit" button.
- txtSurname - points to the "Surname" column header in the Students table.
- txtForename - points to the "Forename" column header in the Students table.

picTeacher – contains school photo of the teacher

lblName – contains the teacher's name and points. upon the loading of the form or the transference of points the label will be refreshed and the points of the teacher and their name will be put in the label

dgdTranferPoints – contains the teacher last point transfers. These will be imported by searching through the dat file and loading all the records that pertain to the teacher

dgdStudents – contains the files students so teachers can find the studID of the students and by clicking on the file of the students the teacher can open up the students records

btnPointsTransfer – when clicked it executes the point transfers or if grdpnts is visible it causes it to become invisible and makes the points transfer boxes visible

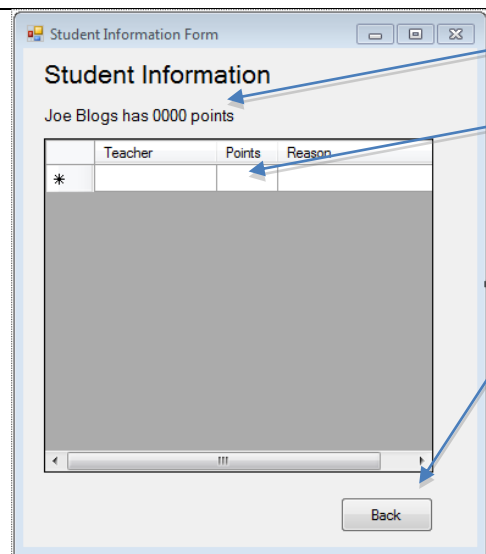
btnPassword – when pressed makes grpPassword visible and changes the buttons text to execute, when pressed a second time if the textboxes containt the right information the password will be changed and grpPasswprd will disappear and the text of the button will revert to change password. Also a message box will pop up telling the teacher that the password has been changed

txtOldPassword – must be the teachers current password

Title: CG2-Green park school reward scheme  
 Centre: Shrewsbury sixth form college (29285)  
 Candidate: Simon Bellows (1054)

txtNewPassword1 – must be at least 6 characters long and be alphanumeric  
 txtNewPassword2 – must be the same as txtNewPassword1  
 grpPassword – contains txtOldPassword , txtNewPassword1 and txtNewPassword2 and is invisible upon opening the document  
 txtForename – can be used search for a student, runs whenever the field is changed  
 txtSurname – can be used search for a student, runs whenever the field is changed  
 btnAdmin – opens up the admin form and hides the teacher form .it is only visible if the staff record says that the member of staff is an admin  
 btnExit – quits the program

## Student information Form

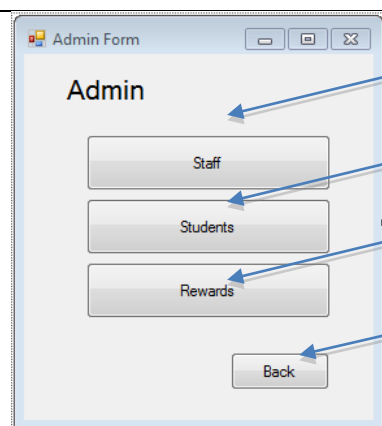


The screenshot shows a window titled "Student Information Form". Inside, there's a label "Student Information" and a text box displaying "Joe Blogs has 0000 points". Below this is a data grid with columns "Teacher", "Points", and "Reason". The first row of the grid contains an asterisk (\*). At the bottom right of the form is a "Back" button. Blue arrows point from the following labels to specific elements in the screenshot:

- LblStudentname** points to the "Student Information" label.
- grdRecord** points to the data grid.
- btnBack** points to the "Back" button.

lblStudentName – contains the students name and points. Upon the loading of the form the information of the student selected will imported.  
 dgdRecord – shows the students records. These will be imported and put into the data grid by searching through the records dat file and finding any that are associated to the student  
 btnBack – closes the window and shows the teacher window again

## Admin Form



The screenshot shows a window titled "Admin Form". Inside, there's a label "Admin" and three buttons: "Staff", "Students", and "Rewards". At the bottom right is a "Back" button. Blue arrows point from the following labels to specific elements in the screenshot:

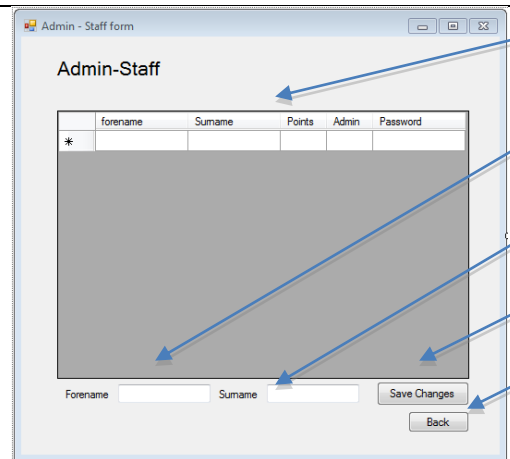
- btnStaff** points to the "Staff" button.
- btnStudents** points to the "Students" button.
- btnRewards** points to the "Rewards" button.
- btnBack** points to the "Back" button.

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

btnStaff – opens up the admin staff form and hides the admin form  
btnStudents – opens up the admin students form and hides the admin form  
btnRewards – opens up the admin rewards form and hides the admin form  
btnBack – closes the window and shows the teacher window again

### Admin – Staff form



The screenshot shows a window titled "Admin - Staff form". Inside, there is a table with columns: forename, Surname, Points, Admin, and Password. The first row has a "\*" in the forename column. Below the table is a large grey rectangular area. At the bottom of the window, there are two text input fields labeled "Forename" and "Surname", a "Save Changes" button, and a "Back" button. Blue arrows point from labels on the right to these components: "grdStaff" points to the table, "txtForename" points to the first cell of the table, "txtSurname" points to the second cell of the table, "btnSave" points to the "Save Changes" button, and "btnBack" points to the "Back" button.

| forename | Surname | Points | Admin | Password |
|----------|---------|--------|-------|----------|
| *        |         |        |       |          |

Forename  Surname  Save Changes Back

grdStaff  
txtForename  
txtSurname  
btnSave  
btnBack

dgdStaff – contains the records of all the staff so that they can be changed and more can be added  
txtForename – can be used search for a member of staff, runs whenever the field is changed  
txtSurname – can be used search for a member of staff, runs whenever the field is  
btnSave – saves changes made to the fields  
btnBack – closes the window and opens the admin window again

Title: CG2-Green park school reward scheme  
 Centre: Shrewsbury sixth form college (29285)  
 Candidate: Simon Bellows (1054)

## Admin – Students form

grdStudent

txtForename

txtSurname

btnSave

btnBack

dgdStudent – contains the records of all the students so that they can be changed and more can be added

txtForename – can be used search for a student, runs when ever the field is changed

txtSurname – can be used search for a student, runs when ever the field is

btnSave – saves changes made to the fields

btnBack – closes the window and opens the admin window again

## Admin – rewards form

grdRewards

txtName

btnSave

btnBack

dgdRewards – contains all the rewards so that they can be changed and more can be added

txtName – can be used search for a reward, runs when ever the field is changed

btnSave – saves changes made to the fields

btnBack – closes the window and opens the admin window again

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

## ***Hardware and Software requirements 2***

### Hardware requirements for visual studio 2010

- Computer that has a 1.6GHz or faster processor
- 1 GB (32 Bit) or 2 GB (64 Bit) RAM (Add 512 MB if running in a virtual machine)
- 3GB of available hard disk space
- 5400 RPM hard disk drive
- DirectX 9 capable video card running at 1024 x 768 or higher-resolution display
- DVD-ROM Drive

### Software requirements

- Windows XP or greater
- Visual studio 2010

## ***Processing stages***      **10**

### **Importing files**

First the program checks if the dat files are present

If they are then sub ends

If not the program finds the csv and imports it to a dat file using its respective structure

Rewards dat as example

If rewards dat present = false

do

    Rewards csv = reward structure

    Rewards structure = rewards dat

Until end of rewards csv

End if

### **Logging on**

Does presence checks for the username and forename

If both present then Checks if username is alpha or numeric. If alpha then it checks the username against the staff id of all the staff till a match found else the code fails and is caught and sub ends. Once match is found then the passwords are compared if they match the person is logged on as a member of staff. If numeric then it checks the username against the student id of all the students until a match is found or the code fails and is caught and sub ends. Once match is found then the passwords are compared if they match the person is logged on as a student. If neither then the username is reported incorrect

If username is number = true

    For 1 to nstudents

        If student.studID = username

            If student.password = password

                Open student form

                Exit sub

        Else

            MessageBox(password incorrect)

    End if

End if

Next

    MessageBox(username incorrect)

Else if username is alpha = true

    For 1 to nstaff

        If staff.staffID = username

            If staff.password = password

                Open staff form

                Exit sub

        Else



Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

```
        MessageBox(Password incorrect)
    End if
End if
Next
    MessageBox(username incorrect)
End if
```

### Changing passwords

The user will enter their old password into 1 text box and there new password twice into two more text boxes. First the program will check that the old password is their current password. It will then check if the new passwords are the same, if they are alpha numeric and if they are more then 5 characters long.

The program will then replace the users password with the new password

Student as example

```
If password1 <> password2
    MessageBox(both new passwords need to be the same)
Exit sub
End if
if length of password 1 < 6
    messagebox(password must be 6 or more characters)
exit sub
end if
if password1 is number = false and is alpha = false then
    If old password = student.password
        Student.password = password1
    Else
        MessageBox(you must input correct old password)
    End if
Else
    MessageBox( password must be alpha numeric)
End if
```

### Adding a student/staff member/reward

when add button on the admin form that pertains to the type that the user wants to add is clicked the corresponding edit form is opened. Once the relevant data such as name points id has been inputted and validated then data is put into the corresponding structure and put into the relevant dat file.

Rewards as example

```
If name = ""
    MessageBox(need to input a name)
Exit sub
End if
If cost is number = false
    MessageBox(need to input a cost as a number)
```

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

```
Exit sub
End if
Rewards.name = name
Rewards.ID = nrewards + 1
Rewards.cost = cost
Put rewards into dat file
```

### Points procedure

When the give points button is clicked the program validates whether the amount has been specified and if it's a number, if a reason has been given, whether the staff member can afford to do it. If give then the points are given to the student and are taken away from the staff member. If take then the points are taken away from the student but without letting the student's points go below 0.

```
If points is number = false
    MessageBox (need to input a number)
    Exit sub
End if
If reason = ""
    Message box (need to input a reason)
    Exit sub
End if
If give
    Student.points = student.points + points
    Staff.points = staff.points - points
    pointsP.studID = student.studID
    pointsP.staffID = staff.staffID
    pointsP.reason = reason
    pointsP.points = points
    put pointsP into dat file
else
    Student.points = student.points - points
    pointsP.studID = student.studID
    pointsP.staffID = staff.staffID
    pointsP.reason = reason
    pointsP.points = - points
    put pointsP into dat file
end if
```

### Buying a reward

When the student clicks on the reward he wants to buy a reward he will be asked if he would like to buy it. If yes then the cost will be subtracted from his points total and a rewardsprocedure record will be created and put into the rewards procedure dat file

```
If Message box ( would you like to buy the reward, yes, no) = yes
    Student.points = student.points - reward.cost
```

Title: CG2-Green park school reward scheme  
Centre: Shrewsbury sixth form college (29285)  
Candidate: Simon Bellows (1054)

---

```
rewardsP.studID = student.studID
rewardsP.rewardID = reward.ID
put rewardsp into dat file
end if
```

## Searching

By presence checks it first works out which of the 4 scenarios it is working out, first searching by both forename and surname, second forename only, third surname only, fourth neither. For the first it uses a for loop checks for each record if the forename begins with the inputted forename and if the surname begins with the inputted surname, if it does then the record is added to the data grid. For the second it's the same but without checking the surname. For the third it's the same but without the forename. If it's the fourth then the grid just loads all the records.

```
if surname is present
    if forename is present
        for 1 to Nsearchdata
            if searchdata.surname begins with surname and searchdata.forename
                begins with forename
                add to data grid
            end if
        next
    else
        for 1 to Nsearchdata
            if searchdata.surname begins with surname
                add to data grid
            end if
        next
    else
        if forename is present
            for 1 to nsearchdata
                if searchdata.forename begins with forename
                    add to data grid
                end if
            next
        else
            for 1 to search data
                add to data grid
            next
        end if
    end if
end if
```

### ***Evaluation criteria***      **3**

Primarily the project will be evaluated on whether is functional and whether it meets the objectives therefore I will judge it on its ability to complete these objectives

- Keeps a consistent house style that keeps colour to a minimum
- Text boxes and buttons are labelled clearly
- Every thing can be easily accessed
- Load time must be under 10 seconds
- Size must not total more than 20 megabytes
- All inputs must be validated so that errors don't often arise
- Logon is password protected and the passwords must be at least 6 characters and be alpha numeric
- Must be able to add and edit rewards, staff and students
- Must not have cost more than £100
- Students must be able to
  - See their points
  - Why they were given points or why they were removed
  - Be able to buy rewards
  - Change their passwords
- Staff must be able to
  - Must be able to see their remaining points
  - Must be able to see the points of all students
  - Must be able to change their passwords
  - Must be able to give and take away points
- Admin must be able to
  - Must be able to add and edit staff records
  - Must be able to add and edit student records
  - Must be able to add and edit rewards
  - Must be able to upgrade a member of staff to admin
  - Must be able to see decrypted passwords so that they can reset them
  - Must be able to give staff more points