9 - Evaluation

Contents

9.1 Effectiveness of the Programming Language

For the development of my system **I decided to use Java** as this programming language was taught to me once I had started my A level Computer Science course, therefore this was the programming language I was most familiar with at the time of starting my project. I also believe that this was the programming language **was most suited for my system as Java is both a procedural and visual** programming language. The original system for which I designed mine on is a sports hall booking system which requires both staff members and customers to make bookings. To make this **system as user friendly and intuitive as possible this would require a Graphical User Interface (GUI)** to appeal to all users whether they are novice or experienced, therefore the use of java allowed me to implement these features. To do **so I utilised the library classes** that Java offers, such **as 'javax.swing' and 'java.awt'** which enables me to access the various components that can be used to create a GUI such **as JButtons, JTextfields and JPanels,** all of which combine to make a system that has **a familiar touch and feel to other modern-day systems,** therefore making this programming language ideal for my system. I also used the **java.time.LocalDate and java.time.format.DateTimeFormatter classes** within my system. Initially I used them to find current date to which I would format into 'dd/mm/yy' format which would be shown in the combo box for dates when making a booking. I again used the **LocalDate class for my bubble sorts on dates.** As I had **stored my dates for bookings as string,** I was then able to use this to **convert them to LocalDate and utilise the isBefore/isAfter methods** to successfully compare my dates to each other and **thus sort them in both ascending and descending order.** This helped make my program efficient and functional as this was the **best implementation** that had found for my sort, but it also allowed me to **successfully meet two of my objectives.** Java also enables the use **of methods and objects** which can be called and used throughout multiple classes. This has allowed me to create **separate class files for each of my entities** for which I was able to create various methods for the entities. These methods would be **called in my main GUI file** which would prompt processes to be run **when components of my GUI were interacted with.** This would therefore **enable the system to be able to respond to user input,** initially allowing navigation but slowly allowing me to build my system **to perform searches and sorts on the click of a button.**

As well as this **Java does not require a specific text editor** to be run alongside with it meaning that I could use any of the **many free text editors** that are available. The one that I used **was Notepad++,** a text editor that allows many features which enhanced the development of my program. These included the **ability to annotate my code,** allowing me to **leave references on certain methods** for me or my peers who were helping me if I had found an issue in my code. It made my **code more readable** and allowed me to **create a suitable structure** which would make the **development process as efficient** as possible. Furthermore, as I had **installed this text editor on my home computer,** I was able to make use of Google Drive and **save any updated source code files** which could be **easily accessed at home,** enabling me to **continue the development of my system easily and smoothly without encountering any compatibility issues.** This therefore was an upside to using java which would only require the Java files **to run on multiple devices** as long as they had **Java and Notepad++ installed.**

Also, the fact that **Java uses a compiler to run** which translates the whole source code altogether, allowed my **debugging process to be smoother,** than it would've been using an interpreted programming language. This is because **I was able to create a batch file** to run the commands required to compile and run the program**. In the event of any errors, the command line would display all of them together allowing me to identify all my bugs at once** and go through each bug alongside my source code and find a solution accordingly. I believe **this helped speed up the process** of my development massively especially throughout the **earlier stages of my development,** as the **building of all my screens and components became quite repetitive** and enduring therefore making me more susceptible to smaller errors, therefore this helped.

9.2 Compare & Contrast with Similar Commercial Systems

'Sports Booker' is an online booking system for sports activities which allows many modern functions that my system doesn't. **Similarly, to mine it allows the booking of a sports activity** at their facilities and uses customer registration meaning **the customer will have to log on** to the system **to access and manage their bookings. My system can match the basic functions** of the system allowing the main requirements to be achieved. However, my system doesn't include some of the features on their system such as **an online integrated system**. This allows the system to **take online bookings from customers as well as staff from any internet connected device.** This is a huge difference in comparison to my system as it allows the **user to have easier access to all their bookings** from anywhere at any time. This was **not included in my system as it was a limitation due to the scale of my project** that my system **would remain offline.** This would therefore **prevent my system from being accessed** from anywhere **resulting in portability issues** which this commercial system doesn't have.

This system also looks **much more professional than mine in** terms of its design. It provides **access to a calendar to manage the bookings** being made by users so that they can easily **be viewed in a simple yet organised way**. This would mean that it is **more user friendly and intuitive compared to mine** as my system stores the bookings into a table of data and requires a separate input from the user to sort by dates. This was **something that I could have implemented into my system had I been more experienced** with the use of Java at the time of my design. This would've therefore allowed me to implement this idea **offering my users a more visual form of output**.

Another system like mine is one called **'Open Play' which offers a sports-based activity** booking system. My system is s**imilar as this allows each member of staff to have login and full access to all the data** regarding the bookings and customers. This system however does offer **the ability to be used on any phone** or other device as it has **both a website and a phone app**. Therefore, the **app to be easily accessed by users** at any time meaning they would **not require a specific device** to run the system. My system **is the opposite as it requires Java to be installed** on the device and compiled on the system before using it.

This system also **allows automatic communications** between the system and customers. The system **can send automatic reminders and text to customers** on their bookings that have been made. This is useful as they do not have to remember the day and time of their booking as they will be reminded of that in advance **making the human-computer-interaction seamless and very intuitive.** This again became a **limitation of my system** as I was not able to create this feature **due to the time scale and resources of my project.** However, **I could have potentially mimicked this feature** by forming a **similar output process to act as confirmation** of bookings even if it was offline.

Overall my system **can complete the same functions as the commercial systems at the very core.** The system can **effectively make bookings** and **allow users to gain access through logins** to manage them. My system also includes **levels of access for staff and customers** ensuring that only customers can access **their own data only** and **staff can access all the data**. However**, my system falls short in terms of its design and professionalism,** making my system **less user friendly** as **users may not trust** it. The **commercial systems are also able to integrate the system** and its **features onto an online platform** allowing a wider field of access. With this alone the systems can **make online bookings and effectively make payments** through different mediums such a credit or debit card which mine cannot do. This **however largely stems from the limitations of my project** which are **restricted due to the time frame and resources** I have had access to.

9.3 Identify Good Features and Shortcomings

9.31 Good Features

During the development of my system there were many tests that I had run to ensure each process **worked effectively, identify any bugs** in the code and areas which may need refinement. For each of these tests **I recorded whether they worked successfully, providing evidence using screenshots** and command line outputs. However, if they did not, I would identify the issue **and correct it with an explanation of the error** that I had encountered again providing evidence for each of test. Through developmental testing I was able to ensure basic functionality within my system for which **I then moved on to testing the system altogether** instead of each process individually. This involved me **testing each of my objectives** and I was able to identify that **every objective had been successfully met** which ensured that my program was **fully functional.** The system can have **two separate users (staff and customer)** who can each individually **add, delete, search and sort the bookings**. The staff members are allowed access to all the bookings and data stored on the system, however the **customers will only have access to data stored within their specific text file.** This therefore makes the system **relatively secure** as once users log on to the system, they will have **certain access depending on the login they provide**. The system also allows the users to **view bookings and payments of bookings in a table**, which for staff would be all the bookings and for customers the history of their own bookings. **On this screen they can sort the data by name or date** (from oldest to latest bookings) and can also delete certain bookings which will **cancel the bookings and remove the data** from the system. All **off screen objectives worked successfully too** such as the **reading and writing to/from files** which again allowed an **intuitive experience** to the user as they would only **receive useful output.**

Additionally, the **design of my system** is also a strength, as each **screen is simple and clean** to allow the user to understand the purpose of the screen immediately. This is also helped with the **headings that are on every screen clearly outlining what it is for.** The system also has a **familiar look and feel to other systems** like mine making **it easier for users** to understand how it works and to be able to use it effectively. This again keeps the system **quite intuitive and user-friendly** to use.

Furthermore, **I implemented many useful validation techniques** which ranged from **presence, range, type and format checks**. This allowed my system to be able to **effective in dealing with incorrect inputs** from the user to **prevent incorrect data being entered into the text files** when storing data. This therefore helps **maintain the integrity of the system** and ensures that **only meaningful data is stored.** The **validation also helps to provide useful information** to the user especially **if the user accidently enters the incorrect data,** which would prompt an error message on that specific error. This is particularly **useful for novice or new users** on the system **who may be unfamiliar** on how certain inputs should be entered. An **example of this is the input when searching for a booking using a date.** If the user enters an incorrectly formatted date, the system will prompt an error message saying **the input should be in the format 'dd/mm/yy'** this would be very **helpful to all users who were unsure** and allows the system to be **user friendly and effective.**

During my **testing I also ran performance tests** to see how well the system **handled processes under excessive strain and larger data sets** than usually expected. I **ran these on all process which would be required to be completed in a fast** and smooth manner to prevent any delays for the users, **such as the searches, sorts and write to file methods.** For each of these processes I ran **them in typical and extreme conditions** and **the system was able to handle them effectively and efficiently when using them in typical conditions.** The **processes remained functional** and **did not cause the system to crash** or slowdown in any way and **allowed the user to continue using the system seamlessly.** This therefore illustrated **a strength in the way the data was handled** and how the user would not be affected in 'normal' circumstances.

9.32 Shortcomings and Minor Refinements

However, **reflecting on feedback** that I have **gained from my Beta Testing, as well as my own criticisms,** there are areas of my system **which would require refinements**. Firstly, would be a refinement of the **design of my system in terms of layout and logos/headings**. Although it was liked by the beta test users, **being described as 'professional' and 'realistic'** there was also some comments that the font that I had used for the **heading and labels were quite distracting** as they were **not very consistent**. I also believe that **the colour scheme I used on the screens were quite basic** and **could've matched the colours of the actual system** it was designed on, making it look **slightly more professional to** make the **users more comfortable** using the system.

The **processes for which I used to write to file, search and sort** the data of my system could have **also been refined.** Although these processes **remained functional during my testing** this was only under typical conditions. Under **extreme conditions the system was unable to remain functional** instead the system would **freeze for a lengthy time between 30 – 90 seconds** which for the system is a massive cause for concern. This therefore would require a **change in the implementation of these methods** using an approach that is more suited to handling extremely larger sets of data. Although the current approach was still able to successfully meet my objectives, **if scaled up it would not be able to cope as well as it does now, identifying an area of refinement in the future.**

A **final refinement of my current system would be making it more informative for all users**. Again, using my beta test feedback, it was **suggested that I should include help popups for users** who would require help on certain aspects of each screen. Expanding on this idea **I also feel that there should be certain visual indicators on screen for inputs that require correcting**. This could be shown in the **form of red text in the text field** which would **turn black when the input became legal**. As well as that there would be **highlighted text fields following the prompt of an error message** so that the **user is able to clearly identify their mistake** and correct it **without them having to proof read** all their inputs. These little changes **would have huge impacts on the user-friendliness** of the system making it **more intuitive** but also removing any confusion between the user and they **system enabling the human-computer interaction to be as smooth as possible**.

Overall the system **meets all the requirements** and **functions as I had initially designed**. Although there are some aspects of the **processes which could be refined,** the **overall system is very effective in achieving its goals by** ensuring users are able **to comfortably navigate and use the system** with no real guidance, **providing a range of features** which makes it easy for them have **control over their bookings.**

9.4 Significant Potential Improvements

One major improvement to my system would be to **include a calendar as a way to view and search for bookings** (by date only). This would be a useful feature as it **would allow for a more visual means of output and input making the system more user friendly and easier to use**. With this the user would be able **to easily select a date on the calendar** and view bookings on that day **rather than having to view a table of booking**s which are not in date order. This **also wouldn't require the user to manually sort and search by date** from the search screen as the **bookings would already be sorted by date** without the user having to make an input **making it more intuitive**. This could also be **improved further by including a calendar sync**. This was an idea **I had mentioned as a limitation during my discussion** and would be suitable for this system. This would **allow customers to sync the dates of booking to their calendar on their chosen device** (i.e. their phone calendar) which enable them to **view their bookings without even logging on** to the system. This would make the system **more efficient as it removes a step from the user's interaction** but also could **potentially cause less stress on the system** as there would be less users on the system at once which could **improve its performance,** making this significant improvement to the system.

Another improvement would be to **improve the security** of the system especially **for staff access** by having them **log onto the system using their fingerprint**. This would make their access to the system **much more secure as this data cannot be lost or stolen by anyone,** ensuring **levels of access are maintained**. This would also be **easier for the staff** as they **would not have to remember any specific details** for them to log in and would prevent any mistakes form them when entering the data as they would no longer need to. This would make the whole **system slightly more efficient from a staff members perspective,** which would help make the system more **useable especially if there is a new staff member.** The **security of the system could also be improved by using encryption** on all data that is written to files. This would be a crucial feature as a **customer details such as card details and contact information are stored in the text files** and could be accessed by anyone on the system. To prevent such details being compromised **I could've created a method which would encrypt the data before writing** it to the text file, so that if the text file was e**ver accessed manually the data wouldn't be readable,** keeping the data stored on the customers **confidential and secure.**

The system **could've been integrated with a cloud-based service** in order to **store the data to form a suitable back up procedure** in the event of any lost data. This would **allow all data that is stored** in a text file to be regularly **uploaded to the cloud securely**. With this implementation the system would be able to **handle a potential crash that resulted in all the data being lost**. Instead of moving to a paper-based system the data stored on the cloud could be **easily downloaded and accessed to have the system return to being functional,** making it a **suitable improvement** as it will store data securely but also provide a backup. Another suitable back-up would be to keep **and store all the systems data off-site on hard drives** in case the other **backup is inaccessible**. This would therefore provide **another possible solution** in the event of system failure; however, this **back-up would be updated less regularly** as this data would be a last resort back-up.

To also improve the system, **I could've implemented a screen to allow the creation of logins** by staff members for new customers. **Currently the only way to add a new login for a customer is to manually** add it to the text file. This would i**nstead allow a screen to allow a new login to be created along with a random password for the customer** to be able to access their booking data. This idea would **also include a screen for the customer to be able to change their password** and other personal details too. This would allow firstly the user's data **to be more secure as they could update the password regularly** but also would allow them to **update their details on the system** ensuring that the data stored in the text files is **up to date maintain the systems integrity**.

9.5 Own Strengths & Weaknesses during Design & Development

The process to **develop my system** required me to **dedicate a lot of my time** to ensuring that my system worked as well as I had initially planned. Throughout the development I **had focussed on achieving my objectives and the functionality** of my system. This was **crucial to me as I wanted the processes to work for both sets of users (Staff and Customers)** to allow a strict separation of access to certain data. This therefore allowed me work hard on **finding the best ways to implement my objectives such as the sorts and searches,** but this also helped ensure that the **file handling of the system was effective too.** This is so that the staff and customers can have access to data that they can such as a customer should only access their own booking/payment file and not be able to see all other bookings on the system. This worked as a **form of security allowing essentially levels of access to** data, but also **maintain confidentiality for other customers payment information** which is important uphold.

The project that I had decided to complete was a **good choice as I knew how the system worked from my own experience**. This allowed me to successfully mimic the system whilst adding **many new and important features to enhance** and modernise the system. However, in terms of the implementation of my **ideas I feel as though I did not set out enough objectives** for me to achieve. Although my system works very well, I feel as though there **could have been more features** which would be useful to the system and enhance the user experience even more. These would include **more variety in the sorts and searches** but also to **apply more than one filter at once** to allow users to find a specific booking. Another **potential objective** would be to **allow encryption of the data** especially when **storing a customer's payment information,** which would require a certain level of security. These **objectives were more than within my capabilities,** yet I did not include them as I **focussed on keeping my system simple and effective** which hindered its potential.

When I **initially started the project** the second stage was the **investigation of the proposed system.** To complete this **required me to obtain a more in depth understanding of the system** that I was planning to create. As my system was based off a **sports centre, Bolton One,** I went to them to ask them **questions of their experiences with the system** that they currently use. After **interviewing a few members of staff,** I had found many **useful insights into the small things** that had either made things easier or more difficult to work with. This allowed me to **understand the system in terms of the user friendliness** and that I should make the system **tailored towards making the user experience more intuitive.** However, from this brief **interview I did not gain a much better insight into the functionality and features** of the system and how they use the system daily. This therefore set me back slightly as **I was not thorough enough in my questioning,** meaning most of my system design was down to my own experience. Furthermore, **my research into similar systems lacked in quality too,** as I did not find a suitable system to compare mine to at the time. This therefore only allowed me to **identify limitations of my system more so than any useful features** that I could have implemented again meaning my investigation fell short of what I was required to find out.

Following my investigation and design, I started to **create the prototype** for my system. This included me creating **the basic framework for the** system implementing the **key features that would be required to work** such as the adding and viewing of bookings. Although I had successfully added a few features into the prototype the **system was not effective in displaying what I had designed.** The structure **of all my screens were very poor** and it was clear **that I had focussed on ensuring the features worked over creating a template for me to build on.** This therefore made **it more difficult to continue with the development,** post-prototype as I had **more work to do due to my poor implementation initially.** This also caused me to **gain some harsh feedback from my peers, prompting me to rethink certain aspects of my design** for the final system.

Once the design was finalised and I had a clear understanding of how to effectively build my system, **I changed the way I approached each section of my project.** I had **initially started, with no real plan** on how I was going to complete each task, **yet along the course of the project I was able to identify** that there ware many **components that would have to be met to create an effective system.** This prompted me to **focus less on the functionality of my processes, allowing me to view the system in a wider perspective and work on integrating higher quality performance, look and feel** into the processes. **This enabled me to work more productively** in the final stages of development ensuring that one part of my system would not take away from another, meaning that **my system was able to be completed to a high quality.**

Overall the **project forced me to change my approach** to building my system as well as **completing work in general** as I had learnt how **to better manage my time** and to **structure the way in which I handled my workload.** Although **I always met the deadlines for each section** of my project there would be times where **I would fall behind causing me to make up for loss time in other areas.** This was **prominent in the early stages of the project** as I feel as though **I am better suited to the practical element** of the project which **included the building and testing of my system.** Therefore, I **struggled to complete the earlier sections** as the work became **quite long and enduring.** However, once I started to **manage my time in a more structured way,** I found that I had **plenty of time to not only finish the section of work but to a higher quality than earlier sections.**

I also learnt many **other skills throughout the project in terms of my programming.** Initially I was able to create a **GUI which would respond to user inputs to form basic** functions such as navigation and reading files. However, the **more research and developing I did I was able to use class libraries effectively utilising their methods** to create my own processes. Even though **there were some limitations** to the methods I had created, which especially struggled to run under extreme conditions, **I was able to successfully meet the main requirements** of the system, which would remain functional and efficient in normal conditions. I also **learnt to reflect on my work** and be **critical of my own ideas** where necessary, when developing my system **which ensured that I would try to find the best possible solution** to any issues that I encountered and **state clearly any major changes** that would be suitable for a future project.

9.6 Future Changes of Approach to Avoid Problems

There were many **aspects of my approach I would change** for this project in the future. As stated earlier **I struggled with completing the earlier stages such as the investigation and design** of the project to a higher standard. Next time **I would ensure that I focus and obtain the necessary information** as these stages are crucial to **allow the project to be completed with little errors**. This would enable me to have **a deeper understanding of the system** I am creating but also the scope of the system too. I would be able to **identify all possible objectives and clearly understand the limitations** that I would face allowing me to build a fully functional system that would contain as **many features each with high performance**. I would also **manage my time slightly better** than I did for this project. I would instead set **out the tasks that I would need to complete each day** to provide myself with a **structured and organised plan** to complete each section of the project well **before any deadlines** to allow me **time to make improvements if required, rather than rushing** some pieces of work and potentially sacrificing the quality while doing so.

Another way I would change my approach **would be to gain more feedback at various stages** of my project. As the system would be used by **a wide range of users,** I feel as though **more in-depth feedback from peers would be key to improving certain elements of my system**. This would allow me to **constantly adapt my ideas** and slowly refine them, so that when implemented the system **would be tailored towards enhancing the user experience** and make the **human-computer-interaction as intuitive as possible.** I would attempt to obtain **feedback on the design section** of my project. This **would firstly allow me to see if the design (screen layout and branding) was appealing** and what the user would like or expect. Secondly it would let me see **if the objectives I had set out to achieve were suitable** for the system and which were not required offering me **an insight into what the user themselves are looking for rather than what I would like to implement.** I also would look to perform **a wider beta test during the testing section** of the project. This again would help me gain wider feedback but also it **would showcase the systems effectiveness** as it would be tested **by real users, whose feedback is the most important.**

Overall, I would look to use an **AGILE method of approach next time.** This would allow me to be **flexible in certain aspects** of my project and enable me to **focus on meeting all necessary requirements** to a higher quality. This methodology would also allow me to **work closely with the end user and refine the design** based on feedback gained throughout the process **enabling me to build an effective and complete system.**