

BeITel

System Documentation



Candidate Number: 5087

Centre Number: 71413

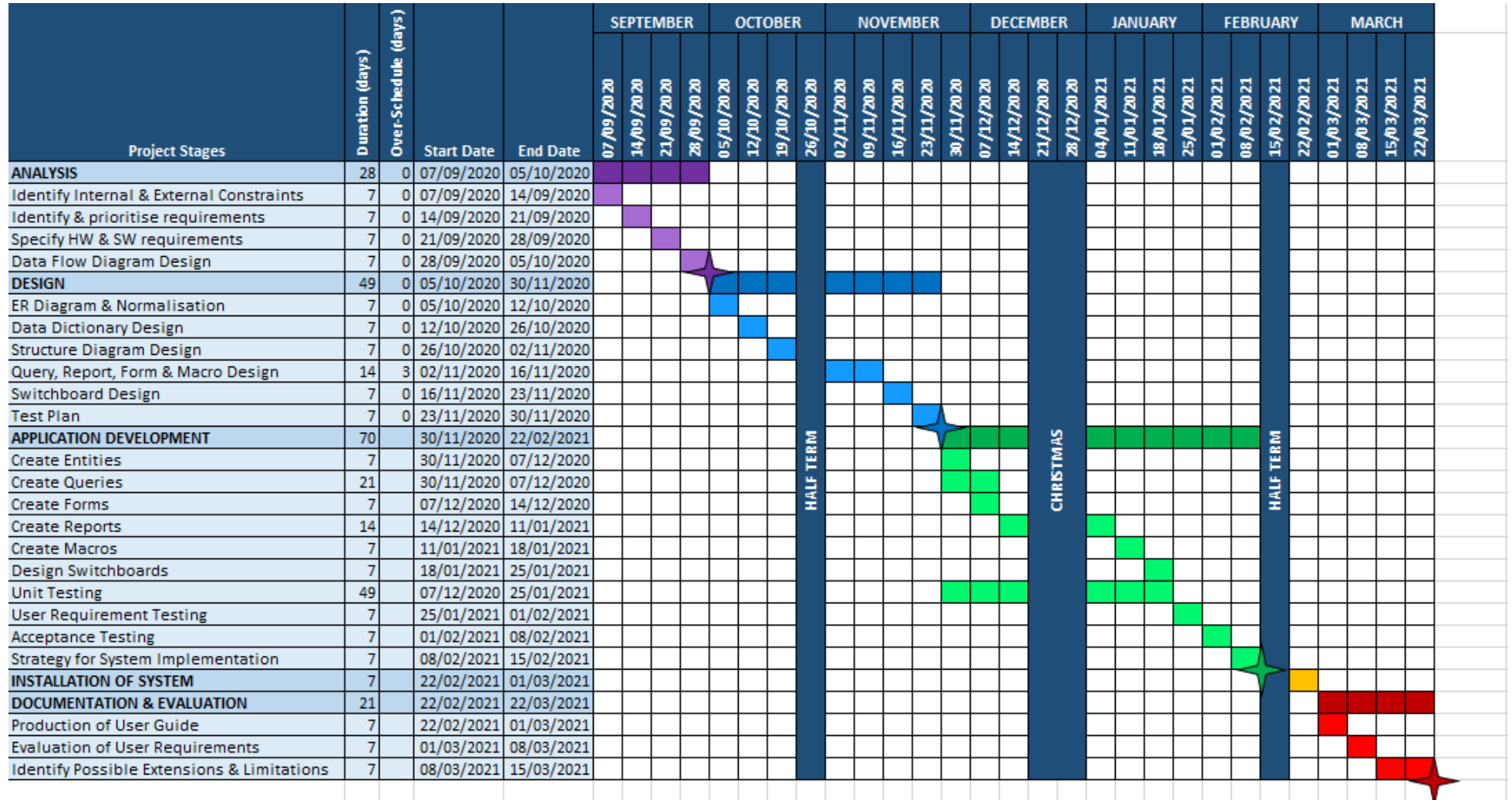
Table of Contents

Chapter 1 – Analysis	1
1.1 Gantt Chart	1
1.2 Project Constraints	1
1.2.1 Internal Constraints	1
1.2.2 External Constraints	2
1.3 User Requirements	5
1.3.1 Inputs	5
1.3.2 Processes	6
1.3.3 Outputs	9
1.4 Essential and Non-Essential Requirements	12
1.4.1 Essential Requirements	12
1.4.2 Non-Essential Requirements	14
1.5 Hardware and Software Requirements	15
1.5.1 Hardware	15
1.5.2 Software	17
1.6 Data Flow Diagram	18
1.6.1 Level 0	18
1.6.2 Level 1	19
Chapter 2 – Design	20
2.1 Normalisation	20
2.2 ER Diagram	22
2.3 Data Dictionary	23
2.3.1 Table 1: tblBookings	23
2.3.2 Table 2: tblEmployee	23
2.3.3 Table 3: tblMedical	24
2.3.4 Table 4: tblMemberMembership	26
2.3.5 Table 5: tblMembers	26
2.3.6 Table 6: tblMembershipLevel	27
2.3.7 Table 7: tblPaymentMethod	27
2.3.8 Table 8: tblPayments	28
2.3.9 Table 9: tblSubscriptionPlan	28
2.3.10 Table 10: tblTreatments	28
2.3.11 Table 11: tblMemberMedical	30
2.4 Structure Diagram	30
2.4.1 Overview Structure	30
2.4.2 Employees Structure	31
2.4.3 Members Structure	32
2.4.4 Bookings Structure	33
2.4.5 Treatments Structure	34
2.5 Form Design	35
2.5.1 Form 1: frmBooking	38
2.5.2 Form 2: frmEmployee	39
2.5.3 Form 3: frmMember	42
2.5.4 Form 4: frmTreatment	47
2.5.5 Form 5: frmMembership	48
2.6 Query Design	49
2.6.1 Query 1: qryAppArchiveBookings	49
2.6.2 Query 2: qryAppArchiveEmployee	49
2.6.3 Query 3: qryAppArchiveMedical	49
2.6.4 Query 4: qryAppArchiveMemberMedical	50
2.6.5 Query 5: qryAppArchiveMemberMembership	50
2.6.6 Query 6: qryAppArchiveMembers	50
2.6.7 Query 7: qryAppArchivePaymentMethod	51
2.6.8 Query 8: qryAppArchivePayments	51
2.6.9 Query 9: qryDelArchivedBookings	51
2.6.10 Query 10: qryDelArchivedEmployee	52
2.6.11 Query 11: qryDelArchivedMedical	52
2.6.12 Query 12: qryDelArchivedMemberMedical	52
2.6.13 Query 13: qryDelArchivedMemberMembership	53
2.6.14 Query 14: qryDelArchivedMembers	53
2.6.15 Query 15: qryDelArchivedPaymentMethod	53
2.6.16 Query 16: qryDelArchivedPayments	54
2.6.17 Query 17: qryBookingLookup	54
2.6.18 Query 18: qryBookingQuarterlyEarnings	54
2.6.19 Query 19: qryBookingsWeekly	55
2.6.20 Query 20: qryEmployeeAddressLookup	56
2.6.21 Query 21: qryEmployeeBookingEarnings	56
2.6.22 Query 22: qryEmployeeBookings	57
2.6.22 Query 22: qryEmployeeLookup	57
2.6.23 Query 23: qryMemberAddressLookup	58
2.6.24 Query 24: qryMemberMedicalRecords	59

2.6.25 Query 25: qryMemberPaymentLookup	59
2.6.26 Query 26: qryMembersBookingSpent	59
2.6.27 Query 27: qryMembersLookup	60
2.6.28 Query 28: qryNoBookingsPerMember	61
2.6.29 Query 29: qryUpdBookingEndTime	61
2.6.30 Query 30: qryUpdCompleteBooking	61
2.6.31 Query 31: qryUpdExpiredMembers	62
2.6.32 Query 32: qryUpdLoyaltyPoints	62
2.6.33 Query 33: qryUpdTemporaryMembers	62
2.6.34 Query 34: qryFrequentlyCancelledBookings	63
2.6.35 Query 35: qryInactiveTemporaryMembers	63
2.6.36 Query 36: qryOutstandingPayments	64
2.6.27 Query 27: qryInactiveMembers	64
2.7 Report Design	66
2.7.1 Report 1: rptEmployeeCard	69
2.7.2 Report 2: rptMembershipCard	71
2.7.3 Report 3: rptReciept	72
2.7.4 Report 4: rptInactiveTemporaryMembers	73
2.7.5 Report 5: rptWeeklyBookings	74
2.7.6 Report 6: rptOutstandingPayments	75
2.7.7 Report 7: rptBookingQuarterlyEarnings	76
2.7.8 Report 8: rptFrequentBookingCancellations	77
2.7.9 Report 9: rptInvoice	78
2.7.10 Report 10: rptInactiveMembers	80
2.8 Macro Design	82
2.8.1 Marco 1: AutoExec	82
2.8.2 Macro 2: mcrArchiveAllRecords	82
2.8.3 Marco 3: mcrArchiveBookingDetails	82
2.8.4 Marco 4: mcrArchiveMemberDetails	82
2.8.5 Marco 5: mcrArchiveEmployeeDetails	83
2.8.6 Marco 6: mcrBackupDatabase	83
2.8.8 Marco 7: mcrUpdateExpiredMembers	85
2.8.9 Marco 8: mcrUpdateBookingEndTime	85
2.8.9 Marco 9: mcrUpdateBookingStatus	86
2.9 Switchboard Design	86
2.9.1 Switchboard Formatting	86
2.9.2 Switchboard Content	87
2.9.3 Switchboard Navigation	90
Chapter 3 – Application Development	91
3.1 Unit Testing	91
3.2 User Requirement Testing	91
3.3 Corrective Action	91
3.2.1 CA#1	91
3.2.2 CA#2	91
3.2.3 CA#3	91
3.2.5 CA#5	92
3.2.6 CA#6	92
3.2.7 CA#7	92
3.4 Acceptance Testing	93
3.5 Strategy for System Implementation	94
3.5.1 Hardware and Software Installation	94
3.5.2 Data Conversion	95
3.5.3 Changeover Method	95
3.5.4 Training	96
3.5.5 Summary	97
Chapter 4 – Evaluation	98
4.1 Evaluation of User Requirements	98
4.1.1 Essential Requirements	98
4.1.2 Non-Essential Requirements	100
4.2 System Limitations	100
4.3 Possible Extensions to the System	100

Chapter 1 – Analysis

1.1 Gantt Chart



1.2 Project Constraints

1.2.1 Internal Constraints

- **Timescale:**

The project will be in development and have a timescale from the 7th of September 2020 to the 22nd of March 2021 that being a length of 28 weeks. This is a hard deadline and therefore is inflexible, the system must be developed and the change over from the old system to the new system completed in this timescale. This will impose restrictions onto the budget as the two are intricately linked.

- **Budget:**

The budget has been set at £15,000 and will only fund the development of the booking area of the system in the Spa and Beauty complex, this will cover the costs of the development of the system, the purchasing of hardware and software for BelTel, the training of BelTel employees, employment of personnel to maintain the system, and the implementation of the system. The budget will directly affect the scope of the project and may lead to some non-essential features not being developed in order to meet the deadline and stay within the budget.

Adaptive maintenance may also be required over the systems lifespan due to changes in the environment within which the system is used in order to ensure the system continues to run smoothly and meets user requirements. This may strongly influence the systems budget as new or changes in external requirements such as legislation may be introduced meaning new functions must be added to the system or overtime internal requirements such as new business requirements may be desired.

Money may also have to be set aside in the budget in order to perform corrective maintenance which is needed to correct errors found in the system once it is in use, a system patch will be issued which is used to update, fix or improve a computer program or its supporting data. Typically, if the problem has been caused by a programming error it will be covered free of charge, however, outside of that it will be covered as part of a rolling contract.

Another form of maintenance that would heavily influence the budget if it is required later down the line is Perfective Maintenance were the systems performance needs to improve, for example this may exclude improving processing speeds by taking advantage of new hardware and software in order to keep the system up-to-date.

- **Skills:**

Only one person has been assigned to the project to complete it within the timeframe given this is due to the fact it is only a small project. However, the member of staff assigned has limited experience in making major projects and therefore will constrain the projects timescale due to inexperience. They will carry out all stages of the System Development Lifecycle from Analysis all the way to Installation and Maintenance applying their knowledge and expertise from the development of other smaller databases in order to address the problem specified by BelTel with an appropriate solution.

- **Hardware & Software:**

The hardware used in-house for development, depending on the location of the developer, will either be a Dell OptiPlex 7020 featuring a 3.3 GHz Quad-Core Processor with 4GB of 1600MHz RAM or a custom-built computer featuring a 3.7GHz 10-Core Processor with 32GB of 3600MHz RAM. This will be more than enough processing power to complete the tasks within the timescale. While the software being used is Microsoft Access, to develop the database and handle its processes, it will also be needed in order to operate the system by BelTel. This will constrain the project as Microsoft Access is the software available for use and the processes developed will have to work within the constraints of the hardware available.

- **Legacy Systems:**

The legacy system must be analysed in order to choose an appropriate changeover method to ensure the new system is properly implemented, this will consequentially have a direct

impact on the timescale as data will have to be converted into a suitable file format for the new system, staff will also need to be trained in the operations of the new system. The current system is paper based therefore time will be needed to key the data into the new computerised system.

- **Programming Language:**

Throughout development, Visual Basic will be used as the high-level programming language of choice, it allows the program to be written using English keywords. It also provides an interactive online help system; it's not just a programming language but it's also got an interactive development environment integrated into it meaning that its graphical user interface is intuitive for management of the program structure and its various types of entities making it very simple to use. Its popularity means it's well documented which makes troubleshooting far easier. It makes use of "IntelliSense" technology which provides real-time, context-sensitive help, which appears right under your cursor by predicting what the user wants to type in without the user typing it in completely. However, this language may also constrain the projects timescale as its much slower than other programming languages since it is a Graphical User Interface based tool with aspects of its graphics requiring a significant amount of space for initial installation and general use therefore utilizing the main memory heavily and it is also exclusive to the Windows Operating system since the programming language was written by Microsoft and therefore cannot be easily transferred to other operating systems since it is their proprietary programming language.

- **Politics:**

In rare cases a project may follow an inappropriate approach to the analysis, design and development of a system due to a new Project Lead wanting to have their own personal touch on the project, or there may be a competitive scenario in the workplace negatively affecting the development of the system however as this system is only being development by one individual without workplace politics, it won't suffer the same fate.

- **Stakeholders:**

The system must comply with BelTel stakeholders wishes however the lack of availability of the stakeholders to produce clarification regarding requirements, can hold the project back as they have insufficient authority also stakeholders who do not have authority to make change requests do so which leads to time constraints. If required BelTel must also make changes to the systems requirements, if requested by BelTel stakeholders, which will inevitably further constrain the budget and add pressure to the time scale provided.

1.2.2 External Constraints

- **Legislation:**

The DPA (Data Protection Act 1998) consists of eight key principles with some additions to said principles by the GDPR (General Data Protection Regulation 2018):

Firstly, it ensures data is lawful and fair, therefore BelTel must ensure that they have legitimate reasons for collecting the data and it must not have any negative effect on the person the data is stored on.

Furthermore, it must be specific for its purpose, BelTel must be open on the reasoning for collecting personal data and what they intend to use it for and it should only be used for the purpose they originally stated. Since GDPR got introduced, data stored on race and religion is now considered sensitive data and can only be stored if it is required for a relevant purpose, BelTel will not be able to store such information as they do not require it.

It must be adequate and only for what is needed, BelTel should only hold on to data for its members and employees that is adequate for their required purpose, they should avoid holding more information than is necessary.

Also, it needs to be accurate and up to date. BelTel will have to take the necessary steps to upkeep data stored on its members and employees, this entails actively ensuring they have the correct data on said individual; BelTel must not use out of date details to contact anyone.

It is not to be kept longer than needed, this means BelTel will have to regularly review the data they retain on both their members and employees, out of date data should be archived or removed from the system this also makes it easier to manage the systems data.

It should take into account people's rights therefore BelTel must ensure that if requested; personnel have the right to access their own personnel data, have inaccurate data changed and as issued by the GDPR now have the right to request information to be deleted from the organisations system.

Data should be kept safe and secure, this entails that BelTel should have a proper security system in place in order to prevent being exposed to security risks. To ensure the security of information it is advised that BelTel should provide adequate training on both the DPA, GDPR and provide logins for their employees such as usernames and passwords in order to make the system more secure by essentially whitelisting who can get onto their systems. BelTel will now require a Data Protection Officer (DPO) as stated by the GDPR as it is now a mandatory role for all companies that collect or process EU citizens' personal data. The DPO will be responsible for educating the company and its staff about compliance, as well as carrying out training for staff on data processing, conducting audits to ensure compliance, etc. They will act as the link between BelTel and the GDPR Supervisory, this all will ensure that personal data is kept safe and secure.

Not be transferred outside the EEA (European Economic Area) meaning that BelTel should not share or transfer information to other countries that don't employ the same standards of data protection, with the introduction of the GDPR, BelTel must have the consent from the person on which the data is stored to transfer their personal information outside of the EEA. BelTel should also provide an Acceptable Use Policy to its employees which is a document detailing constraints and practices that the user will have to agree to and sign in order to be granted access to their network and systems.

CDPA (Copyright, Design and Patents Act 1988) is the UK's current copyright law and covers the rights of creators in order to ensure that they can control the ways in which their material is being used. In order to conform with this law BelTel must ensure it does not use any copyrighted imagery, music or literature within their database without the permission of the original creator of the work. Also they need to ensure they get any original works such as logos copyrighted as well as their company name, such as BelTel® and purchase a proper site license in order to use software across multiple computers on a network legally.

TMA (Trademarks Act 1999) covers the registration and protection of trademarks within the UK. BelTel must not infringe on any trademarked material and to protect their own brand should register for a trademark, they should also register to trademark their own brands in order to protect themselves from other companies using their branding, such as BelTel® Spa & Beauty Complex™.

CMA (Computer Misuse Act 1998) protects personal data by illegalizing any unauthorised access or modification to digitized material with or without further intent of crime. Crimes covered by this law include hacking, computer fraud, blackmail and viruses. To conform to this law BelTel must ensure there are proper security measures in the Database to prevent any authorised access to personal data held within the database, they should also prohibit employees from using personal USBs in work to prevent possible malicious software getting on the network or personal information being compromised. An extra step BelTel could take in order to ensure the security of their systems is to install network monitoring software. Username and Passwords for employee logins would also be required in order to prevent someone such as a customer breaching the database internally.

HASAWA (Health and Safety at Work Act 1974) outlines the legal obligations of employers to protect the health, welfare and safety of all their employees at work. In relation to the database this law requires the safe operation, (including maintenance) of equipment and systems, therefore BelTel must ensure staff are properly trained in order to carry out operations of the systems. The software should also be designed in such a way as to minimise mouse clicks by implementing “Tab Stops” which makes it far easier for employees with mobility issues to enter data into a form as they no longer have to use the mouse to click onto the next field.

DDA (Disability Discrimination Act 1995) bans discrimination by the employers against disabled employees and requires them to be accommodated. BelTel must avoid breaching the regulations of this law by implementing accessibility measures into the software such as captions or text alternatives for non-text content, operable by making use of the functionality of a keyboard and ensuring the navigation is easy to follow and finally understandable by making text clear for the visually impaired.

PCI DSS (Payment Card Industry Data Security Standard) rules out a set of requirements with the intent to ensure that all companies that are in contact with the processing, storage and transmission of credit card information maintain their security. As BelTels booking system requires a payment method they are legally obligated to ensure the security of credit card processes by having an appropriate and reliable security system in place that is password protected to ensure their payment details aren't maliciously breached.

- **Cultural Differences:**

The database being developed is only intended to be operated in one country and therefore will only be designed to specifically meet the requirements of that one location. The language used will be English and will make use of Pounds as its currency. This will not restrict or conflict with the development of the database.

- **Industry Standards:**

It is important to conform to the industry standards so the system is easily interpreted and operated if staff were to commence work on a project mid-way through for database objects:

Using the “**Leszynski Reddick**” naming convention as it is the most commonly used naming convention, which consist of prefixes to indicate the type of object. E.g. tblMembers, qryWeeklyBookings, etc.

Applying “**ISOs Ergonomics of human-system interaction**” to the database which will cover the underlying basis of BelTels accessibility standards. It covers themes such as user preference settings, accessibility adjustments, alternative input options and general visual outputs, all of which will be applied to BelTels database in order to ensure it can operate without any discrimination towards their employees, this also ties in nicely to Disability Discrimination Act 1995.

Consistent **User Interface** styles within BelTels system by utilizing components such as Colour, Text, Iconography and Patterns. Extra functionality can also be added such as tooltips to identify the user interfaces elements.

Within **Visual Basic**, its only appropriate for BelTel to use a proper naming scheme when designating a controls name to ensure that, when coding, it's easy to interpret what that control does from its name, for example if it's to do with the gender of an individual it should clearly say “*Gender*”. Similar to Leszynski Reddicks naming convention, consistent prefixes should be applied to the names of controls within the code in order to identify the its type, for instance, command would be “cmd” and a combo box would be “cbo”. Another standard that can be applied to Visual Basic is involving the codes structure. The highest level statements should be identified by one tab following this each “block” should be indented with an additional tab. Comments should also be left in the code after a double slash to tell the programmer what this line of code actually does since it's sometimes hard to tell from a brief glance.

Data Types should be formatted appropriately using an input mask, for example the data type “*Date and Time*” should be formatted in relation to the desired outcome, if they wish to only display the hour and minutes it should be formatted as “*h:mm*” while if they only wish to display the year and month it should be “*yyyy-mm*”. Other formats can be selected from a dropdown list as well such as Long and short Dates. This also applied to other data types such as Number, Currency, etc. Text data types can also

1.3 User Requirements

1.3.1 Inputs

#	Data Stored	Data Source	Current Method of Input	Why Needed?
I1	Member Details TypeOfMember, Title, Firstname, Surname, Gender, PhoneNumber, EmailAddress, Address, Town, County, Postcode, DateOfBirth, SubscriptionPlan, MedicalConditionName, Restrictions, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, Expiration Year, SecurityNumber Members can have multiple Medical Conditions (Optional) and Payment Methods	Paper Form	Members are given a membership form where they will specify their detailed including medical conditions if relevant and payment details. The form will then be handed in to an employee which will then check everything is entered and store the form.	Members are required to register as a member to use the facility within the Spa and Beauty Complex, this is also required to book a treatment.
I2	Employee Details Title, Firstname, Surname, Gender, DOB, PhoneNumber, EmailAddress, Address, Town, County, Postcode, Department, Poistion, NationalInsuranceNumber, Biograph	Paper Form	Employees are given an employment form which they will fill it, once completed the form will be given to Human Resources were it will be check to ensure all the necessary details have been entered.	The company must have an employee's details in order to properly register them as an employee working for the company, contact details are also required to supply the employee with the appropriate information such as rotas, etc.
I3	Booking Details MemberFirstname, MemberSurname, TreatmentName, EmployeeFirstName, EmployeeSurname, GroupSize, DateOfBooking, TimeStart, EndTime. Members can have multiple bookings.	Online Booking	Customers can make bookings over the phone or in person with the receptionist where she will enter in the details specified by the customer and designated a suitable timeslot for the booking.	Details must be stored on bookings in order for them to operate in an orderly, organised format.

1.3.2 Processes

#	Processes	Frequency	Justification
P1	Add, Edit and Delete Member Records	Adhoc	<p>When a Member joins, they fill in a membership form which they then sign to confirm that the details are correct. This form is checked by a member of staff and they are assigned and a unique Member ID.</p> <p>To comply with the GDPR and DPA the member can request to amend or update information on their records, they can also request that their information be deleted.</p>
P2	Add, Edit and Delete Payment Records	Adhoc	<p>Within the Membership form the member will enter their payment details in order to pay for their membership costs and treatment bookings, there can be multiple Payment Records per individual of which can be entered in at a later date.</p> <p>This will also make processing payments far quicker as they will already be saved on the system.</p> <p>GDPR and DPA enforces that the member has the right to request that their information be amended or updated and can even request information to be deleted.</p>
P3	Add, Edit and Delete Medical Records	Adhoc	<p>When Members fill in their Membership form, they have the option to declare any medical information which will be stored for staff who are carrying out treatments to consult. Members can have multiple medical condition records.</p> <p>GDPR and the DPA requires that the member must be able to amend or update their information as well as request it to be deleted.</p>
P4	Add, Edit and Delete Employee Records	Adhoc	<p>When an employee is employed, they must fill out an employment form which they will enter in their details. This will be reviewed by Human Resources and stored for use, they will then be assigned a unique identifier.</p> <p>To comply with the GDPR and DPA the employee can request to amend or update information on their records, they can also request that their information be deleted.</p>
P5	Add, Edit and Delete Booking Records	Adhoc	<p>A Member will ask an employee to book in a treatment, the employee will then take their booking information, find an</p>

			<p>available slot and enter in the information.</p> <p>If the member needs to change the booking times or what type of treatment, etc. they will be able to ask for an employee to change or edit the details.</p> <p>If the booking gets cancelled the record will be deleted.</p>
P6	Calculate Employee Salary	Adhoc	Needed for financial planning for year ahead.
P7	Calculate Booking Total	Adhoc	The Sub-Totals is calculated to find out the booking cost before discount and is also needed to calculate the booking total, so staff know how much to charge the customers. Then the eligible discount is calculated in order to find the discount amount that can be applied to the booking in order to calculate the booking total for adding to an invoice. Finally, the booking total cost is calculated using these previous calculations as the Member must know the total cost that they are paying for the booking, also it is needed to find out weekly and quarterly booking sums.
P8	Calculate Booking End Time	Daily	This is calculated in order to find out the time at which a booking ends using the duration of the treatment and the specified start time of a booking. So the system can work out when a member of staff is available to start the next booking.
P9	Calculate Member Loyalties	Adhoc	For each booking they complete a member will have one loyalty point added.
P10	Remove or Archive Cancelled Bookings	Weekly	In order to ensure the system doesn't store any unnecessary or redundant data, cancelled bookings will be archived so staff can still see which members frequently cancel bookings.
P11	Archive temporary members after 28 days	Monthly	Members have the option to choose a temporary trial period which lasts for 28 days, after which their membership will not renew if they decide not to. It is then only appropriate to archive this information as it is no longer required.
P12	Archive inactive members	Yearly	Inactive members are not required in working storage as they are no longer using their membership, they will be archived if they have been inactive for 2 years.
P13	Archive completed bookings	Daily	Completed bookings are no longer required to be stored with the active bookings and will be removed from working storage, therefore it will be archived in case records are needed to be searched and retrieved. Need to be

			held specifically for three months / one quarter.
P14	Archive past employees	Weekly	After an employee hands in their two-week notice, their information will no longer be required to be stored on the system and its only appropriate to archive their data to ensure the database response time is not slowed.
P15	Backup Database	Daily	Daily Incremental Backups will be performed each day so data can be recovered in the unlikely event of a disaster, as detailed in the Disaster Recovery Plan
P16	Search for given members.	Adhoc	Retrieves Member data when a specific name is entered.
P17	Search for given employee	Adhoc	Retrieves Employee data when a specific name is entered.
P18	Automatically apply a discount to a Members Booking depending on their Membership Level. <ul style="list-style-type: none"> • Standard 0% • Deluxe 5% • Premium 10% 	Daily	When a member has selected a membership level their discount should be applied to their account automatically without the Staff having to apply it individually, this is to reduce possible errors when it comes to the discount amount.
P19	Automatically apply a 10% discount on bookings with a cost over £200	Daily	When an order reaches over two hundred pounds an automatic discount of ten percent should be applied as an incentive.
P20	Retrieve top earning employees	Daily	BeITel need to see the top earning employees when it comes to treatments to take into consideration when offering promotions or giving bonuses.
P21	Calculate Employee Treatment Commission.	Daily	Employees are offered a five percent commission on each treatment they do in order to incentivise them to prioritise treatments and work harder.
P22	Calculate weekly treatment income.	Weekly	To see how well the Spa Complex and which treatment earned the most in a single week.
P23	Protected entry with use of employee logins (Username and Passwords)	Adhoc	To comply with the General Data Protection Regulation 2018 and Data Protection Act 1998 to increase security.
P24	Validation for Members DOB to ensure they are older than 18.	Adhoc	To become a member and book in for a treatment BeITel requires that they must be at least 18 years or older.
P25	Validate number of members in a group booking to ensure there is more than or equal to 5 or less than or equal to 10.	Adhoc	BeITel requires that in order to book as a group you must have at least 5 in a party with a maximum of 10 in order to utilise the treatments provided.
P26	Calculate cost of Membership depending on the type and subscription plan select	Adhoc	In order to ensure Members pay the correct amount depending on their membership and subscription plan, this is done to ensure BeITel isn't over charging or undercharging their members.

1.3.3 Outputs

#	Output	
O1	Receipt	
	Purpose:	So the member has a “proof of purchase”, it is also useful in refunding a booking however not required.
	Frequency:	Adhoc
	Format:	Printed
		Grouping: N/A
		Sorting: N/A
	Calculations:	Cost multiplied by the GroupSize to figure out the Sub-Total and then subtracted by the EligibleDiscount in order to get the Total.
	Inputs:	BookingID, GroupBooking, DateOfBooking, TimeStart, EndTime, Status, MemberFirstname, MemberSurname, PhoneNumber, EligibleDiscount, TreatmentName, Cost, Duration, EmployeeFirstname, EmployeeSurname.
O2	Invoice	
	Purpose:	An invoice is issued so the member knows how much they must pay.
	Frequency:	Adhoc
	Format:	Printed
		Grouping: BookingID
		Sorting: N/A
	Calculations:	Cost multiplied by the GroupSize to figure out the Sub-Total and then subtracted by the EligibleDiscount in order to get the Total.
	Inputs:	BookingID, GroupBooking, DateOfBooking, TimeStart, EndTime, Status, MemberFirstname, MemberSurname, PhoneNumber, EligibleDiscount, TreatmentName, Cost, Duration, EmployeeFirstname, EmployeeSurname
O3	Membership Renewal Notification	
	Purpose:	This is automatically texted or emailed to a member to alert them that their membership is running out soon, this is to remind them to renew it.
	Frequency:	Adhoc
	Format:	Digital
		Grouping: N/A
		Sorting: N/A
	Calculations:	The expiration date of their membership subtracted by a certain amount of days in order to calculate a date at which the notification will be sent.
	Inputs:	MemberID, Title, Firstname, Surname, PhoneNumber, Email Address, ExpirationDate
O4	Membership Card	
	Purpose:	Acts similarly to a gym card, grants access to all the facilities members are allowed to use for free in the Spa & Beauty complex.
	Frequency:	Adhoc
	Format:	Printed
		Grouping: N/A
		Sorting: N/A
	Calculations:	N/A
	Inputs:	MemberID, Title, FirstName, Surname, Gender, DOB, TypeOfMember, DateJoined, ExpirationDate
O5	Employee Card	
	Purpose:	Used as proof that they work for Beltel, also used to access staff specific areas and systems.
	Frequency:	Adhoc
	Format:	Printed
		Grouping: N/A

		Sorting:	N/A
		Calculations:	N/A
	Inputs:	EmployeeID, Title, Firstname, Surname, Position, Gender, DOB, Department, Status, DateOfEmployment	
O6	Inactive Members Report		
	Purpose:	This is done in order so that BelTel can make an attempt to contact them in order to give the member an incentive to become an active member again, promotional details would be posted out, etc.	
	Frequency:	Monthly	
	Format:	Printed	
		Grouping:	ExpirationDate
		Sorting:	Surname Ascending
		Calculations:	N/A
	Inputs:	MemberID, Title, FirstName, Surname, PhoneNumber, EmailAddress, Address, Town, County, Postcode, TypeOfMember, MembershipLevel, SubscriptionPlan, ExpirationDate	
O7	Reservation Confirmation		
	Purpose:	This is to confirm that the reservation has all the appropriate details and the member is happy to go through with the booking.	
	Frequency:	Adhoc	
	Format:	Digital or Printed	
		Grouping:	N/A
		Sorting:	N/A
		Calculations:	Sub-Total is the treatment cost multiplied by the group sized and Total which is the Sub-Total subtracted by Eligible Discount.
	Inputs:	BookingID, GroupBooking, DateOfBooking, TimeStart, EndTime, Status, MemberFirstname, MemberSurname, PhoneNumber, EligibleDiscount, TreatmentName, Cost, Duration, EmployeeFirstname, EmployeeSurname.	
O8	Booking Refund		
	Purpose:	When a booking is cancelled BelTel must be able to process a refund if they are eligible.	
	Frequency:	Adhoc	
	Format:	Digital	
		Grouping:	N/A
		Sorting:	N/A
		Calculations:	Sub-Total is the treatment cost multiplied by the group sized and Total which is the Sub-Total subtracted by Eligible Discount.
	Inputs:	PaymentMethodID, PaymentID, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, ExpirationYear, SecurityNumber, BookingID, GroupSize, TreatmentCost, MemberID, Title, Firstname, Surname	
O9	Booking Reminder		
	Purpose:	This is a text or email sent to the member in order to remind them that they have an upcoming booking.	
	Frequency:	Daily	
	Format:	Digital	
		Grouping:	N/A
		Sorting:	N/A
		Calculations:	The date of their booking subtracted by a certain amount of days in order to calculate a date at which the notification will be sent.
	Inputs:	BookingID, DateOfBooking, MemberID, Title, Firstname, Surname, PhoneNumber, EmailAddress	
O10	List of people with outstanding payments		

	Purpose:	This is required in order for BelTel to contact those who still owe money.		
	Frequency:	Monthly		
	Format:	Printed		
		Grouping:	Type Of Member	
		Sorting:	Surname Ascending	
	Calculations:	N/A		
Inputs:	MemberID, Title, Firstname, Surname, PhoneNumber, EmailAddress, Address, Town, County, Postcode, AmountOutstanding, TypeOfMember, MembershipLevel, SubscriptionPlan			
O11	Most popular treatments report			
	Purpose:	To give BelTel data on which treatment is the most popular and least popular in order to decide what promotions and resources to allocate to each.		
	Frequency:	Monthly		
	Format:	Printed		
		Grouping:	Status	
		Sorting:	TreatmentName Ascending	
Calculations:	N/A			
Inputs:	TreatmentID, TreatmentName, Duration, Description, Cost, BookingID, Status			
O12	Report of Temporary Members that haven't renewed their membership			
	Purpose:	In order to send offers to temporary members to give them an incentive to renew their membership.		
	Frequency:	Monthly		
	Format:	Printed		
		Grouping:	TypeOfMember	
		Sorting:	Surname Ascending	
Calculations:	N/A			
Inputs:	MemberID, TypeOfMember, Title, Firstname, Surname, PhoneNumber, EmailAddress, Address, Town, County, Postcode			
O13	Employee weekly bookings rota			
	Purpose:	This will give each employee their weekly treatment bookings so they will know where they must be what treatment their doing and what time it's at.		
	Frequency:	Weekly		
	Format:	Printed		
		Grouping:	Surname	
		Sorting:	TreatmentName Ascending	
Calculations:	N/A			
Inputs:	EmployeeID, Title, Firstname, Surname, Department, Position, BookingID, GroupBooking, GroupSize, DateOfBooking, TimeStart, EndTime, TreatmentName, Duration			
O14	Quarterly Bookings report			
	Purpose:	To find out how well Treatments performed over the quarter, this will show what times of the year are most popular for members to book-in.		
	Frequency:	Quarterly		
	Format:	Printed		
		Grouping:	Status	
		Sorting:	TreatmentName Ascending	
Calculations:	N/A			
Inputs:	TreatmentID, TreatmentName, Duration, Description, Cost, BookingID, Status			
O15	Members Who Frequently Cancel Bookings			
	Purpose:	In order to find out which members are possible liabilities, therefore this set of information would allow BelTel to possible give them some form of booking cooldown.		
	Frequency:	Monthly		

	Format:	Printed
	Grouping:	Status
	Sorting:	Surname Ascending
	Calculations:	N/A
	Inputs:	BookingID, Status, MemberID, Title, Firstname, Surname
O16	Quarterly Treatment Profit Graph	
	Purpose:	This will show visually how well each treatment preformed at a financial standpoint in that given quarter.
	Frequency:	Quarterly
	Format:	Printed
	Grouping:	DateOfBooking
	Sorting:	TreatmentName Ascending
	Calculations:	Cost multiplied by the GroupSize to figure out the Sub-Total and then subtracted by the EligibleDiscount in order to get the Total which will be needed to figure out the total amount of earnings in a given quarter.
O17	Inputs:	TreatmentID, TreatmentName, Duration, Description, Cost, BookingID, DateOfBooking, GroupSize, MemberID, EligibleDiscount
	Yearly Finance Graph (Line Graph)	
	Purpose:	This will visually display BelTels peak times within a given year in order to orientate their promotions and deals as well as allocating resources.
	Frequency:	Yearly
	Format:	Printed
	Grouping:	DateOfBooking
	Sorting:	TreatmentName Ascending
	Calculations:	Cost multiplied by the GroupSize to figure out the Sub-Total and then subtracted by the EligibleDiscount in order to get the Total which will be needed to figure out the total amount of earnings in order to add points onto the line graph.
	Inputs:	TreatmentID, TreatmentName, Duration, Description, Cost, BookingID, DateOfBooking, GroupSize

1.4 Essential and Non-Essential Requirements

1.4.1 Essential Requirements

#	Requirement	Reason	I, P, O #
E1	Add, Edit and Delete Member Records	Entire database is dependent on the management of these records of information and is therefore required as a basis.	P1, I1
E2	Add, Edit and Delete Payment Records		P2, I1
E3	Add, Edit and Delete Medical Records		P3, I1
E4	Add, Edit and Delete Employee Records		P4, I2
E5	Add, Edit and Delete Booking Records		P5, I3
E6	Backup Systems	To ensure BelTel do not lose their data in the event of a disaster.	P15, I1-3
E7	Archive Records	To comply with the GDPR as records are not to be kept for longer than needed; Also in order to free up working memory so the system can respond to data and retrieve requests quicker.	P11-14, I1-3
E8	Invoice	This lets the customer know how much their total is for their booking	O2, I3

E9	Receipt	To ensure that the customer has a proof of purchase.	O1, I3
E10	Update Member status after trial period of 28 days	As stated by the end user, a member must have the option to become a temporary member which lasts a total of 28 days. After this period, they can update their membership to be permanent or else cancel it. The status needs updated to reflect their decision.	P11, I1
E11	Members must be 18 or older	The end user states that to become a member you must be at least 18 or older, this is more than likely due to the facilities used within the Hotel and it's also focused on a more mature audience.	P24, I1
E12	Must be a member to book a treatment	In order to book in a treatment, you must first become a member as stated by the end user.	P1, I1
E13	Group bookings have a minimum size of 5 and maximum size of 10	The end user wishes to allow clients to have the option to select a group booking, of which must have at least 5 other members and at a maximum 10.	P25, I3
E14	Update loyalty point for each booking	To incentivise members to book in treatments more frequently, as discussed with the end user, points will be allocated which will go towards a discount.	P9, I3
E15	Password protected with employee logins	To ensure the system is secure and in order to comply to legal regulations such as the GDPR/DPA, CMA and PCI DSS	P23, I2
E16	Apply Membership fees depending on membership level	Members will have to pay a monthly, quarterly or yearly sum of money towards their membership depending on which level they have selected.	P26, I1
E17	Update discounts on bookings	Bookings must have applied discounts on them such as the loyalty discount, the membership discount and the total cost of the booking may also call for a discount to be applied. Deluxe members will receive a 5% Discount and Premium members will receive a 10% discount; furthermore any order over £200 will receive a 10% discount and members can obtain loyalty points which can further contribute.	P18, P19, P9, I1, I3
E18	Employee weekly booking rota	This must be created so employees can look up what date they are designated to a treatment as well as the start time and duration.	O13, I2
E19	Search for given members	This is to retrieve Member data when a specific name is entered, so that BelTel can look up their information if required.	P16, I1
E20	Search for given employees	This is to retrieve Employee data when a specific name is entered, so that BelTel can look up their information if required.	P17, I2
E21	Calculate Employee Treatment Commission	Employees are offered a five percent commission on each treatment they do in order to incentivise them to prioritise treatments and work harder.	P21, I1
E22	List of people with outstanding payments	This is required in order for BelTel to contact those whom still owe money.	O10, I1
E23	Membership Card	Acts similarly to a gym card, grants access to all the facilities members are allowed to use for free in the Spa & Beauty complex.	O4, I1

E24	Temporary Members who have not renewed their membership report	In order to send offers to temporary members to give them an incentive to renew their membership.	O12, I1
E25	Quarterly Bookings Report	To find out how well Treatments performed over the quarter, this will show what times of the year are most popular for members to book-in.	O14, I3
E26	Members who frequently cancel bookings report	In order to find out which members are possible liabilities, therefore this set of information would allow BelTel to possibly give them some form of booking cooldown.	O15, I1, I3
E27	Calculate weekly treatment income	To see how well the Spa Complex and which treatment earned the most in a single week.	P22, I3

1.4.2 Non-Essential Requirements

#	Requirement	Reason	I, P, O #
N1	Membership Renewal Notification	This is automatically texted or emailed to a member to alert them that their membership is running out soon, this is to remind them to renew it.	O3, I1
N2	Employee Card	Used as proof that they work for BelTel, also used to access staff specific areas and systems as an alternative method to entering their login details or code.	O5, I2
N3	Reservation Confirmation	This is to confirm that the reservation has all the appropriate details and the member is happy to go through with the booking.	O7, I3
N4	Inactive Members Report	This is done in order so that BelTel can make an attempt to contact them in order to give the member an incentive to become an active member again, promotional details would be posted out, etc.	O6, I1
N5	Booking Reminder	This is a text or email sent to the member in order to remind them that they have an upcoming booking.	O9, I3
N6	Popular Treatments Report	To give BelTel data on which treatment is the most popular and least popular in order to decide what promotions and resources to allocate to each.	O11, I3
N7	Quarterly Treatment Profit Graph	This will show visually how well each treatment performed at a financial standpoint in that given quarter.	O16, I3
N8	Yearly Finance Graph	This will visually display BelTels peak times within a given year in order to orientate their promotions and deals as well as allocating resources.	O17, I3
N9	Retrieve top earning employees	BelTel need to see the top earning employees when it comes to treatments to take into consideration when offering promotions or giving bonuses.	P20, I2

1.5 Hardware and Software Requirements

1.5.1 Hardware

The computer used for the development of the system is more than capable of running a database system for BelTel. It has the following specifications.

Developing Computer Specification:

Item	Specification
RAM	Corsair Vengeance Pro 32GB (2 x 16GB) DDR4 3600MHz
CPU	Intel i9-10900k 10 Core, 20 Threads, 3.7GHz, 20MB Cache
MOBO	Asus ROG MAXIMUS XII Apex
	Ports 10 x USB 3.2, Intel 2.5G Ethernet
GPU	AFOX GeForce RTX 2060S 8GB DDR6
SSD	Samsung 860 EVO 2Tb, Read 550MB/s, Write 520MB/s
Monitor(s)	1 32" LED Asus ROG Strix XG32VQ 2560x1440 DisplayPort @ 144Hz
	2 24" LED Acer K242HL 1920x1080 HDMI @ 75Hz

Recommended Minimum Computer Specification:

Dell OptiPlex 7020 Small Form Factor Desktop Computer		
Item	Purpose	Minimum Specification Suggested
RAM	Random Access Memory is volatile memory which can be written to and read from, its used for holding data on application software the user is currently working with and also stores a portion of the operating system in it, it's considered as "Main Memory".	8GB (2 x 4GB) DDR3 1600MHz 8GB of RAM is more than enough memory to store the necessary information required during BelTels operation, since it has a speed of 1600MHz it will also be fast and efficient.
CPU	The CPU processes everything the computer does and is responsible for carrying out the fetch/ execute cycle. The clock speed within a CPU is timing signals that are provided by an internal clock, it determines how many computer instructions are carried out (executed) each second, for example a CPU running at 3GHz will be able to process three thousand million instructions per second. A multi-core processor will be able to perform tasks simultaneously as each core can handle one instruction at a time.	Intel i5-4590 Quad Core, 3.3GHz Turbo
Cache	Cache is like RAM however it is a lot faster as it is integrated into the processor, its used to work through demanding tasks quickly as it solves the problem of time delays between RAM and the processor by acting as a middle man between the two.	6MB Cache
Ports	The Universal Serial Bus ports provide a high-speed serial connection to most types of USB peripheral devices, its industry standard makes it common place for peripheral devices to use and is therefore required for the recommended list of peripherals. The High-Definition Multimedia Interface port transmit video and audio data from the computer to the output device, in this case it	4 x USB 2.0, 1 x HDMI 4 ports have been selected to allow peripherals to be connected to the computer. Only one HDMI port is required as their will only be a single monitor connected to the computer.

	is essential to display video onto a monitor in order to operate the system.	
GPU	The GPU (Graphical Processing Unit) is an electrical component found within a computer, it can either be dedicated or integrated into the processor. It is solely responsible for the manipulation and alteration of memory in order to increase the speed at which the computer can present an image. Its main purpose is to handle the computer's graphics, visuals and videos.	Integrated Intel HD Graphics 4600
HDD	The Hard Disk Drive is a secondary storage device, it has a large storage capacity which makes it suitable storing the operating system and data files and it's cheap on a cost per megabyte compared to other storage devices, therefore, it will be used for bulk storage of BelTels operational data and their related backups of subsequent databases.	500GB 7200rpm SATA 500GB will be more than enough storage space and with a speed of 7200rpm, data retrieval and processing should be at a sufficient speed.
Monitor	The monitor will display the computer's user interface along with different software using a series of RGB LED's in order to give the illusion of an image, the resolution of the screen determines the quality of the image displayed and is therefore directly correlated with the quality. Every monitor will also come with a predetermined refresh rate which will define how smooth motion is displayed.	24" LED 1920x1080 @ 60Hz Its medium sized in order to save room and make information easier to see but also to cater to visually impaired colleagues. 1920x1080 is standard HD resolution which is more than capable of being used to operate the computer system. It has a 60Hz refresh rate which is also standard among monitors and makes movement on screen seem smooth rather than jittery, it makes the computer look and feel much faster and means it won't distract employees.

Recommended Peripherals:

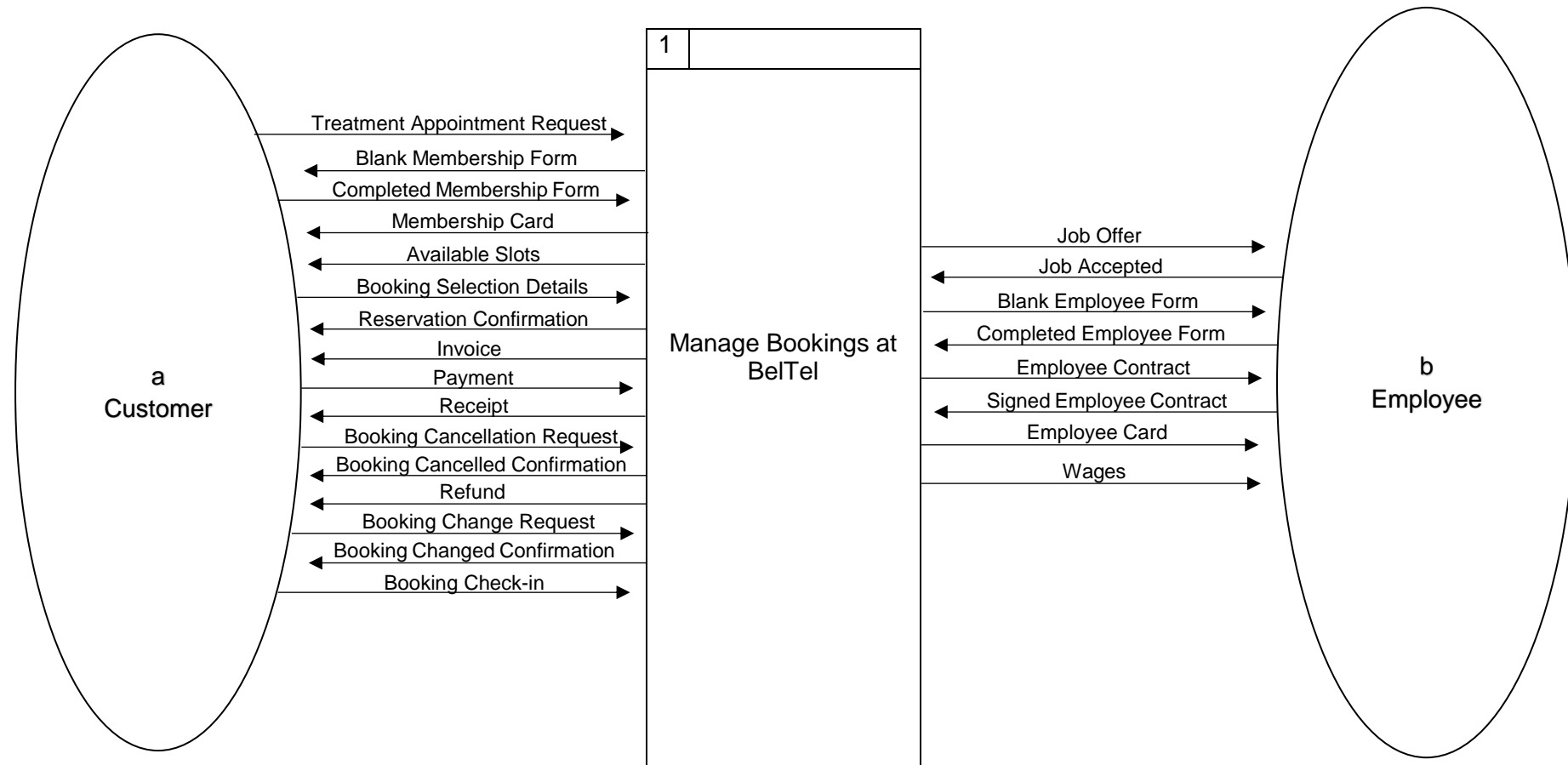
Item	Purpose
Payment Terminal	This will allow BelTels members to pay with a card that isn't already saved onto the system.
QWERTY Keyboard	This is an essential input device and is required to enter in text.
Mouse	This is an input device required to manoeuvre the cursor/ pointer on the screen to navigate the system and select different windows, icons and menus.
Printer	This is required to print off different forms such as the Membership Registration form and the Employee Registration form, it's also useful to print off POS (Point of Sale) and finance sheets, furthermore, it gives the consumer a choice of receiving a physical invoice on the day of booking rather than a digital one via email. Another incredibly useful aspect of purchasing this peripheral is that it will be able to print off MIS Reports so BelTel can be constantly aware of their sales and financial standpoint.

1.5.2 Software

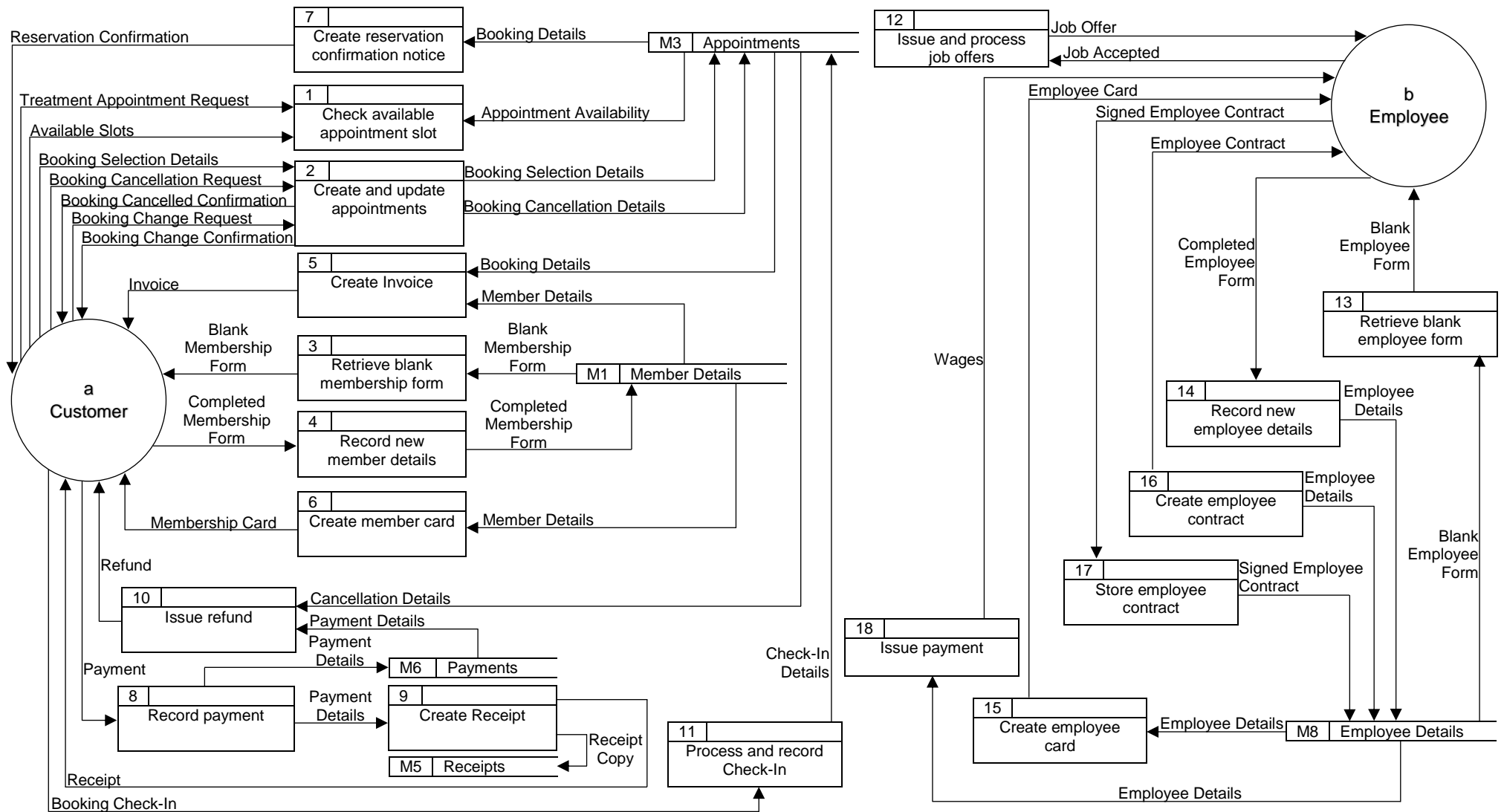
Item	Purpose	Software Package Suggested
Operating System	The operating system is responsible for the management of all operations and resources of a computer, it acts as an interface between the hardware, application software and the end user.	Windows 10 Enterprise Windows 10 is the most suitable OS for BelTel as it's the most widely used and is common place in most business and personal computers, therefore most users will already be familiar with its graphical user interface, the OS is also well supported. Windows 10 Enterprise has been suggested over other editions due to its built-in security advantages.
Application Software	In order to operate and maintain the system this application is required, it helps analyse large amounts of data, its validation, processing and output and comes with many other features such as visual tools for designing queries, printing off forms, etc.	Microsoft Access 2016
	Outlook is an email client and is recommended in order to establish proper communication between colleagues, personalised emails can also be used for colleague logins such as jkirkpatrick081@ni.beltel.com .	Microsoft Outlook 2016
	Excel is a spreadsheet program which will allow BelTel to compile their financial data in an appropriate format as well as other appropriate information such as yearly debit sheets and yearly credits.	Microsoft Excel 2016
	Anti-Virus software also known as anti-malware is a utility used to prevent, detect and remove malware, its primary purpose is to protect the computer from viruses. This program will add an extra layer of protection on top of Windows built in security systems in order to defend BelTels systems and member/ employee information.	Avast Anti-Virus

1.6 Data Flow Diagram

1.6.1 Level 0



1.6.2 Level 1



Chapter 2 – Design

2.1 Normalisation

1NF

To achieve 1st Normal Form, the repeating attributes must be removed, in this case, "MemberID, PaymentMethodID, TypeOfMember, Title, Name, Gender, PhoneNumber, EmailAddress, Address, DateJoined, Loyalty, EligibleDiscount, DOB, Photograph Directory, Archive, MedicalID, ConditionName, MemberMedicalID, Restrictions, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, ExpirationYear, SecurityNumber, PaymentID, AmountOutstanding, Status, MembershipLevelID, MembershipLevel, Description, Cost, MemberMembershipID, PaidDate, ExpirationDate, SubscriptionPlanID, SubscriptionPlan, CostMultiplier" into a new entity called "**MEMBER-PAYMENT**". This new entity now requires a primary key, so "MemberID" and "PaymentMethodID" was allocated as the Composite Key. To create a relationship between both entities as required by 1st Normal Form, "MemberID" was added as a foreign key to "**MEMBER-BOOKING**". Finally, to ensure all fields are atomic, "Name" was split into two attributes, that being "Firstname" and "Surname", "Address" was also split into "Address", "Town", "County" and "Postcode".

MEMBER-PAYMENT (MemberID, PaymentMethodID, TypeOfMember, Title, Firstname, Surname, Gender, PhoneNumber, EmailAddress, Address, Town, County, Postcode, DateJoined, Loyalty, EligibleDiscount, DOB, Photograph Directory, Archive, MedicalID, ConditionName, MemberMedicalID, Restrictions, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, ExpirationYear, SecurityNumber, PaymentID, AmountOutstanding, Status, MembershipLevelID, MembershipLevel, Description, Cost, MemberMembershipID, PaidDate, ExpirationDate, SubscriptionPlanID, SubscriptionPlan, CostMultiplier)

MEMBER-BOOKING (BookingID, MemberID*, EmployeeID, TreatmentID, Title, Firstname, Surname, Gender, DOB, PhoneNumber, Email Address, Address, Town, County, Postcode, Department, Position, NatornallInsuranceNo, DateOfEmployment, Status, ContractedHours, Wage, Biograph, Photograph Directory, Archive, TreatmentName, Duration, Description, Cost, GroupBooking, GroupSize, DateOfBooking, TimeStart, EndTime, Status)

2NF

To achieve 2nd Normal Form, you must find all the entities which consists of a composite key, in this case it is "**MEMBER-BOOKING**" and "**MEMBER-PAYMENT**" you then go on to examine each attribute not part of the primary key, in the process deciding whether it is dependent on the primary key or just part of it. Following this you must then remove the attributes that were only partially dependent on the primary key and create new entities containing the attributes that were partially dependent, for this database the new entities created are: "**EMPLOYEE**" with the following attributes "EmployeeID, Title, Firstname, Surname, Gender, DOB, PhoneNumber, Email Address, Address, Town, County, Postcode, Department, Position, NatornallInsuranceNo, DateOfEmployment, Status, ContractedHours, Wage, Biograph, Photograph Directory, Archive" with "EmployeeID" being the primary key, a copy of "EmployeeID" was also made in "**MEMBER-BOOKING**" to link the entities together. "**MEMBER-MEDICAL**" which stores the attributes "MemberMedicalID, MedicalID*, Restrictions". "MemberMedicalID" has been established as the primary key with the foreign key in this table being "MedicalID*", with the purpose of this table being to establish a link from "**MEMBER-MEDICAL**" to "**MEDICAL**" in order to allow members to have multiple restrictions linked to their established medical conditions. "**MEDICAL**" with the attributes "MedicalID, MemberID, ConditionName" with "MedicalID" being the attribute used as the primary key and the foreign key being allocated as "MemberID" to link "**MEDICAL**" to "**MEMBER**". "**MEMBER-PAYMENT**" which holds the attributes "PaymentID, MemberID*, AmountOutstanding, Status", its foreign key is "MemberID*" its purpose is to create a relationship between "**MEMBER**" and "**PAYMENT**", it also allows members to have multiple records in "**PAYMENT**" and also to store member specific information. "**PAYMENT**" with the attributes "PaymentMethodID, PaymentID*, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, ExpirationYear, SecurityNumber" with "PaymentMethodID" being the primary key for this entity and "PaymentID*" being the foreign key to establish a connection between "**PAYMENT**" and "**MEMBER-PAYMENT**". "**MEMBER-MEMBERSHIPLEVEL**" holding the attributes

“MemberMembershipID, MemberID*, MembershipLevelID*, SubscriptionPlanID, SubscriptionPlan, CostMultiplier, PaidDate, ExpirationDate” with the primary key assigned being “MemberMembershipID” and the foreign keys assigned in this entity being “MemberID*, MembershipLevelID*”. This table was created in order to establish a relationship between **“MEMBER”** and **“MEMBERSHIPLEVEL”** but also to allow a member to select a membership level without having data about its description and cost in each Member record, this prevents data redundancy. **“MEMBERSHIPLEVEL”** with the attributes “MembershipLevelID, MembershipLevel, Description, Cost” with “MembershipLevelID” being the primary key.

EMPLOYEE (EmployeeID, Title, Firstname, Surname, Gender, DOB, PhoneNumber, Email Address, Address, Town, County, Postcode, Department, Position, NatornalInsuranceNo, DateOfEmployment, Status, ContractedHours, Wage, Biograph, Photograph Directory, Archive)

MEMBER-BOOKING (BookingID, MemberID*, EmployeeID*, TreatmentID, TreatmentName, Duration, Description, Cost, GroupBooking, GroupSize, DateOfBooking, TimeStart, EndTime, Status)

MEMBER (MemberID, TypeOfMember, Title, Firstname, Surname, Gender, PhoneNumber, EmailAddress, Address, Town, County, Postcode, DateJoined, Loyalty, EligibleDiscount, DOB, Photograph Directory, Archive, MedicalID, ConditionName, MemberMedicalID, Restrictions, MembershipLevelID, MembershipLevel, Description, Cost, MemberMembershipID, PaidDate, ExpirationDate, SubscriptionPlanID, SubscriptionPlan, CostMultiplier)

MEMBER-PAYMENT (PaymentID, MemberID*, AmountOutstanding, Status)

PAYMENT (PaymentMethodID, PaymentID*, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, ExpirationYear, SecurityNumber)

3NF

Finally, to reach 3rd Normal Form, you must examine each entity to determine if any attribute within them is dependent on a non-key field, for this database “SubscriptionLevelID” in the entity **“MEMBER-MEMBERSHIP”** and “TreatmentID” in **“MEMBER-BOOKING”** were non-key attributes with each having other attributes dependent on them. Furthermore, you must then remove the attributes dependent on these non-key fields and put them into new entities with a copy of each attribute in the original entity as the foreign key, the new entites created were **“TREATMENTS”** with the following attributes “TreatmentID, TreatmentName, Duration, Description, Cost”. The designated primary key for this entity is “TreatmentID”, also making a copy of “TreatmentID” in **“MEMBER-BOOKING”** in order to create a relationship between the entities as these attributes are partially dependent. **“SUBSCRIPTIONPLAN”** with the following attributes “SubscriptionPlanID, SubscriptionPlan, CostMultiplier”. The primary key being allocated under the attribute “SubscriptionPlanID”, a copy of this attribute was also left in **“MEMBER-MEMBERSHIP”** as a foreign key in order to establish a relationship between these entities.

EMPLOYEE (EmployeeID, Title, Firstname, Surname, Gender, DOB, PhoneNumber, Email Address, Address, Town, County, Postcode, Department, Position, NatornalInsuranceNo, DateOfEmployment, Status, ContractedHours, Wage, Biograph, Photograph Directory, Archive)

TREATMENT (TreatmentID, TreatmentName, Duration, Description, Cost)

MEMBER-BOOKING (BookingID, MemberID*, EmployeeID*, TreatmentID*, GroupBooking, GroupSize, DateOfBooking, TimeStart, EndTime, Status)

MEMBER (MemberID, TypeOfMember, Title, Firstname, Surname, Gender, PhoneNumber, EmailAddress, Address, Town, County, Postcode, DateJoined, Loyalty, EligibleDiscount, DOB, Photograph Directory, Archive)

MEMBER-MEDICAL (MemberMedicalID, MedicalID*, Restrictions)

MEDICAL (MedicalID, MemberID*, ConditionName)

MEMBER-PAYMENT (PaymentID, MemberID*, AmountOutstanding, Status)

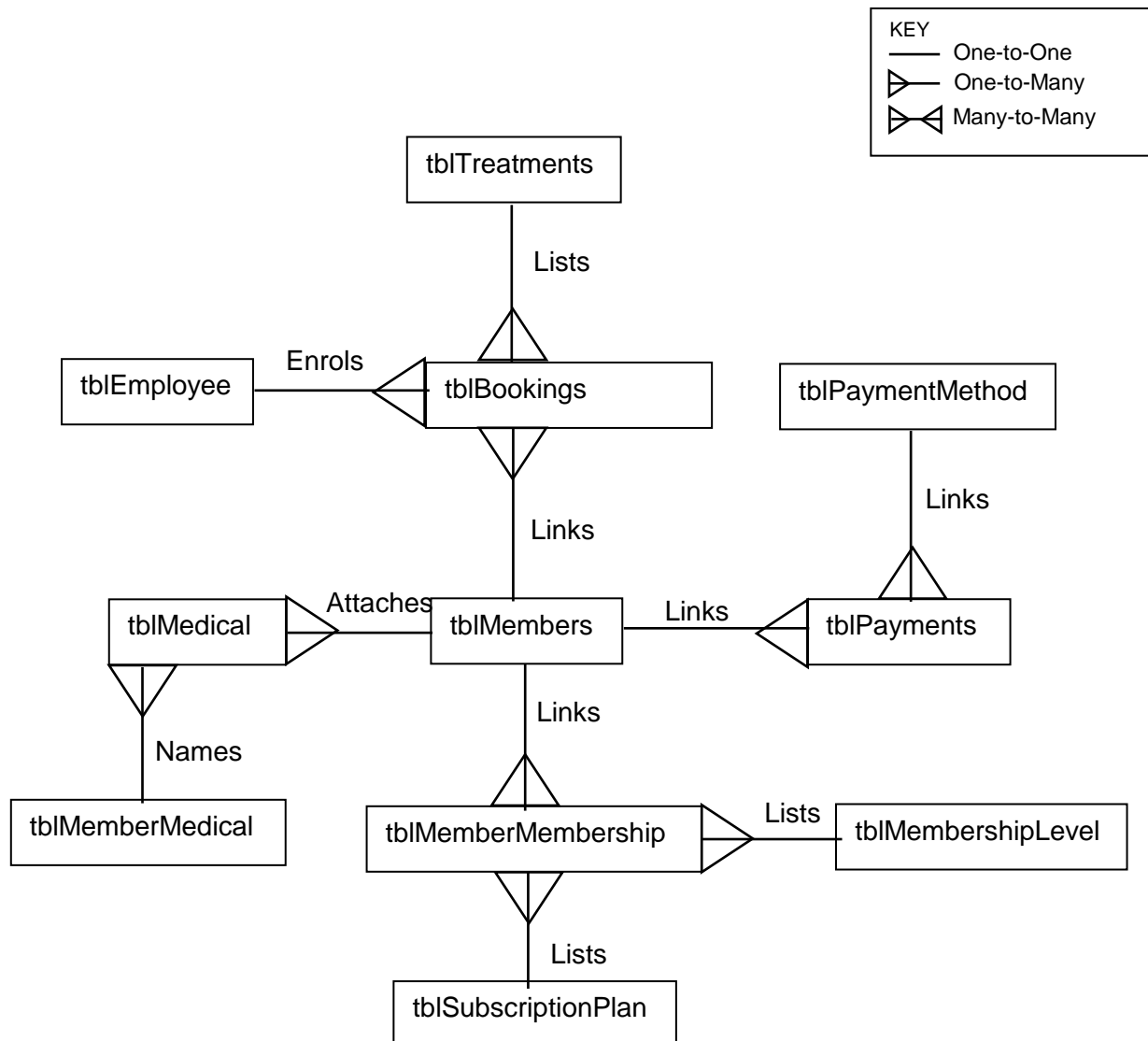
PAYMENT (PaymentMethodID, PaymentID*, PaymentMethod, PaymentType, CardNumber, ExpirationMonth, ExpirationYear, SecurityNumber)

MEMBER-MEMBERSHIP (MemberMembershipID, MemberID*, MembershipLevelID*, SubscriptionPlanID*, PaidDate, ExpirationDate)

MEMBERSHIPLEVEL (MembershipLevelID, MembershipLevel, Description, Cost)

SUBSCRIPTIONPLAN (SubscriptionPlanID, SubscriptionPlan, CostMultiplier)

2.2 ER Diagram



2.3 Data Dictionary

Key

R	Required	Y	Yes	X	No Duplicates	PK	Primary Key	IM	Input Mask	DV	Default Value
I	Index	N	No	*	Duplicates OK	EM	Error Message	VR	Validation Rule		

2.3.1 Table 1: tblBookings

Purpose of Entity: To store data on each booking.

Entity Name:	tblBookings	Primary Key:	BookingID	Foreign Key:	TreatmentID	Related To:	tblTreatments
					EmployeeID		tblEmployee
					MemberID		tblMembers

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	BookingID	✓	X	AutoNumber	Long Integer		1
	MemberID	✓	*	Number	Long Integer	VR: Lookup from tblMembers	2
	EmployeeID	✓	*	Number	Long Integer	VR: Lookup from tblEmployee	6
	TreatmentID	✓	*	Number	Long Integer	VR: Lookup from tblTreatments	2
	GroupBooking			Yes/No		DV: No	✓
	GroupSize			Number	Integer, 0DP	DV: 1 VR: Between 5 And 10 EM: Must enter group size between 5 and 10.	5
	DateOfBooking	✓		Date/Time	Short Date	DV: Now() VR: <Now() EM: Must not be a date before now.	11/09/20
	TimeStart	✓		Date/Time	Short Time		10:00
	EndTime			Date/Time	Short Time		10:50
	Status	✓		Short Text	9	VR: Lookup to include Booked, Completed and Cancelled EM: Must select option from list	Booked

2.3.2 Table 2: tblEmployee

Purpose of Entity: To store data on each employee.

Entity Name:	tblEmployee	Primary Key:	EmployeeID	Foreign Key:	None	Related To:	tblBookings
--------------	-------------	--------------	------------	--------------	------	-------------	-------------

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	EmployeeID	✓	X	AutoNumber	Long Integer		1
	Title			Short Text	4	VR: Lookup to include Mr, Mrs, Miss, Dr and Rev	Mrs
	Firstname	✓		Short Text	30	EM: "Must enter Firstname"	Andrew
	Surname	✓		Short Text	30	EM: "Must enter Surname"	Gallagher
	Gender	✓		Short Text	1	VR: Lookup to include M and F EM: Please select option from list	M
	DOB	✓		Date/Time	Short Date	VR: <=DateAdd("yyyy",-16,Date())	17/02/1998

						EM: "Must be atleast 16 years old to work."	
PhoneNumber	✓		Short Text	12		IM: 00000&000000 EM: "Must enter your mobile number in a format of five numbers following with a space then six numbers."	07514 602345
EmailAddress			Hyperlink			VR: Like "**@*.*"	agallagher775@gmail.com
Address	✓		Short Text	50		EM: Must enter address.	1 Ferndale Grove
Town	✓		Short Text	20		EM: "Must enter town."	Glengormley
County	✓		Short Text	11		EM: "Must select county." VR: Lookup to include Antrim, Armagh, Down, Fermanagh, Londonderry and Tyrone	Antrim
Postcode	✓	*	Short Text	8		IM: >LL09&0LL EM: "Postcode must be enter with two letters following with one number or an additional number depending on your post code, following with one number and then two letters."	BT36 5AP
Department			Short Text	23		VR: Lookup to include Indoor Swimming Pool, Outdoor Swimming Pool, Sauanas, Steam Rooms, Indoor Heated Loungers, Hot Tubs and Treatments EM: Must select department from list.	Indoor Swimming Pool
Position	✓		Short Text	18		VR: Lookup to include Customer Assistant and Manager EM: Must select position option from list.	Customer Assistant
NationalInsuranceNo	✓		Short Text	13		IM: >LL&00&00&00&L EM: "Must enter national insurance number starting with two capital letters, followed by six numbers and then another letter."	SD 25 56 45 D
DateOfEmployment	✓		Date/Time	Short Date		DV: =Now() VR: Not >Now() EM: "The date joined cannot be a date in the future."	20/10/2019
Status	✓		Short Text	9		VR: Is Not Null VT: "Must select the type of employee." VR: Lookup to include Part-Time and Full-Time	Part-Time
ContractedHours	✓		Number	Long Integer, ODP		DV: 8 VR: Not <8 EM: "Must enter contracted hours greater or equal to eight."	16
Wage	✓		Currency	2DP		DV: 8.73 VR: >=8.73 EM: "Must enter a wage greater than eight pound seventy three pence."	£8.73
Biograph			Long Text				"I Love Fishing!"
Photograph Directory			Short Text	255		VR: Like "**.*"	E:\Year 14\Digital Technology\Mrs Nevin\Portfolio Data\Employee Photos\Keith-Mahood.jpg
Archive			Yes/ No				✓

2.3.3 Table 3: tblMedical

Purpose of Entity: To store data on each booking.

Entity Name:	tblMedical	Primary Key:	MedicalID	Foreign Key:	MemberID	Related To:	tblMemberMedical
							tblMembers

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	MedicalID	✓	X	AutoNumber	Long Integer		1
	MemberID	✓	*	Number	Long Integer		8
	ConditionName	✓		Short Text	30	EM: Must enter name of condition.	Nut Allergy

2.3.4 Table 4: tblMemberMembership

Purpose of Entity: To link tblMember to tblMembershipLevel and tblSubscriptionPlan without a Many-to-Many relationship to prevent data redundancy and also to store data relating to payment and expiration dates.

Entity Name:	tblMemberMembership	Primary Key:	None	Foreign Key:	MemberID	Related To:	tblMembershipLevel
					MembershipLevelID		tblMember
					SubscriptionPlanID		tblSubscriptionPlan

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	MemberMembershipID	✓	X	AutoNumber	Long Integer		1
	MemberID	✓	*	Number	Long Integer	EM: Must enter Members ID	1
	MembershipLevelID	✓	*	Number	Long Integer	EM: Must enter Members, Membership Level ID VR: Lookup from tblMembershipLevel	4
	SubscriptionPlanID	✓	*	Number	Long Integer	EM: Must enter Members, Subscription Plan ID VR: Lookup from tblSubscriptionPlan	2
	PaidDate	✓		Date/ Time	Short Date	VR: Not >Now() EM: Paid date cannot be in the future.	15/09/20
	ExpirationDate	✓		Date/ Time	Short Date	VR: Not <Now() EM: Expiration date cannot be in the past.	15/10/20

2.3.5 Table 5: tblMembers

Purpose of Entity: To store data on each member.

Entity Name:	tblMembers	Primary Key:	MemberID	Foreign Key:	None	Related To:	tblMedical
							tblBookings
							tblPayments
							tblMemberMembership

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	MemberID	✓	X	AutoNumber	Long Integer		1
	TypeOfMember	✓		Short Text	13	VR: Lookup to include Adult, Couple, Concessionary and Temporary EM: "Must select the members type."	Adult
	Title			Short Text	4	VR: Lookup to include Mr, Mrs, Miss, Dr and Rev EM: Must select option from list.	Mr
	Firstname	✓		Short Text	30	EM: "Must enter Firstname."	John
	Surname	✓		Short Text	30	EM: "Must enter Surname."	Smith
	Gender	✓		Short Text	1	EM: Select option from the list. VR: Lookup to include M and F	M
	PhoneNumber	✓		Short Text	12	IM: >00000&000000 EM: "Must enter your mobile number in a format of five numbers following with a space then six numbers."	07544 395678
	EmailAddress	✓		Hyperlink		VR: Like "***@*.**" EM: "Must include an @ symbol and periods"	jsmith32@yahoo.com
	Address	✓		Short Text	50	EM: "Must enter address."	3 Kilburn Park
	Town	✓		Short Text	20	EM: "Must enter town."	Larne
	County	✓		Short Text	11	EM: "Must select county."	Antrim

						VR: Lookup to include Antrim, Armagh, Down, Fermanagh, Londonderry and Tyrone	
	Postcode	✓	*	Short Text	8	IM: >LL09&0LL EM: "Postcode must be enter with two capital letters following with two numbers, a space then a number followed by another two capital letters."	BT25 7SZ
	DateJoined	✓		Date/Time	Short Date	DV: =Now() VR: <=Now() EM: "The date joined cannot be a date in the future."	06/05/2020
	Loyalty			Number, Long Integer	0DP	DV: 0	0
	EligibleDiscount			Number, Double	Percent	DV: 0	10%
	DOB	✓		Date/Time		VR: <=DateAdd("yyyy",-18,Date()) EM: "You must be 18 or older to register as a member."	15/07/2000
	Photograph Directory			Short Text	255	VR: Like ".*\."	E:\Year 14\Digital Technology\Mrs Nevin\Portfolio Data\Employee Photos\Keith- Mahood.jpg
	Archive			Yes/No			Yes

2.3.6 Table 6: tblMembershipLevel

Purpose of Entity: To store data on each type of membership level.

Entity Name:	tblMembershipLevel	Primary Key:	MembershipLevelID	Foreign Key:	None	Related To:	tblMemberMembership
--------------	--------------------	--------------	-------------------	--------------	------	-------------	---------------------

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	MembershipLevelID	✓	X	AutoNumber			1
	MembershipLevel	✓		Short Text	15	EM: Must enter Membership Level Name.	Standard
	Description	✓		Long Text			"Standard membership grants access to the facilities."
	Cost	✓		Currency	2DP	VR: Not <=0 EM: Must specify cost of Membership Level.	£7.99

2.3.7 Table 7: tblPaymentMethod

Purpose of Entity: To store data on members payment methods.

Entity Name:	tblPaymentMethod	Primary Key:	PaymentMethodID	Foreign Key:	PaymentID	Related To:	tblPayments
--------------	------------------	--------------	-----------------	--------------	-----------	-------------	-------------

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	PaymentMethodID	✓	X	AutoNumber	Long Integer		1
	PaymentID		*	Number	Long Integer		4
	PaymentMethod	✓		Short Text	16	VR: Lookup to include Mastercard, Visa, American Express and Discover EM: Please selected a supported payment method.	Visa

	PaymentType	✓		Short Text	6	VR: Lookup to include Credit, Debit and Charge EM: Must select a payment type.	Credit
	CardNumber	✓		Short Text	19	IM: >0000&0000&0000&0000 EM: Must enter card number with four numbers followed with a space, repeat this four times.	4462 6354 7023 1107
	ExpirationMonth	✓		Date/Time	MM	VR: Not >=Now() EM: Must enter valid month expiration date.	02
	ExpirationYear	✓		Date/Time	YY	EM: Must enter valid year expiration date.	21
	SecurityNumber	✓		Short Text	3	IM: Password VR: Not Like "[a-z]" EM: Must enter Security Number.	***

2.3.8 Table 8: tblPayments

Purpose of Entity: To link tblMember to tblPaymentMethod without a Many-to-Many relationship to prevent data redundancy and also to store data relating to outstanding payments and the cards status.

Entity Name:	tblPayments	Primary Key:	PaymentID	Foreign Key:	MemberID	Related To:	tblMembers tblPaymentMethod
--------------	-------------	--------------	-----------	--------------	----------	-------------	--------------------------------

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	PaymentID	✓	X	AutoNumber			1
	MemberID	✓	*	Number		VR: Lookup from tblMember	6
	AmountOutstanding			Currency	2DP		£17
	Status	✓		Short Text		VR: Lookup to include Current, Secondary EM: Please select option from list	Current

2.3.9 Table 9: tblSubscriptionPlan

Purpose of Entity: To store data on different types of subscription plans available.

Entity Name:	tblSubscriptionPlan	Primary Key:	SubscriptionPlanID	Foreign Key:	None	Related To:	tblMemberMembership
--------------	---------------------	--------------	--------------------	--------------	------	-------------	---------------------

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	SubscriptionPlanID	✓	X	AutoNumber			1
	SubscriptionPlan	✓		Short Text	9	EM: Must enter Subscription Plan Name.	Monthly
	CostMultiplier	✓		Number, Long Integer	0DP	DV: 0 EM: Must specify the amount to multiply the cost by in relation to time e.g. One Quarter = 3 or Year = 12	3

2.3.10 Table 10: tblTreatments

Purpose of Entity: To store data on different types of treatments available to book.

Entity Name:	tblTreatments	Primary Key:	TreatmentID	Foreign Key:	None	Related To:	tblBookings
--------------	---------------	--------------	-------------	--------------	------	-------------	-------------

PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	TreatmentID	✓	X	AutoNumber			1
	TreatmentName	✓		Short Text	15	VR: Len([TreatmentName])>=4	Facial

					EM: Must enter a treatment name and be a minimum of 4 characters.	
	Duration	✓	Date/Time	ShortTime	EM: Must enter the total duration.	01:00
	Description		Long Text			Facials are skin-care treatments
	Cost	✓	Currency	2DP	VR: Not <=0 EM: Must specify treatment price.	£30.00

2.3.11 Table 11: tblMemberMedical

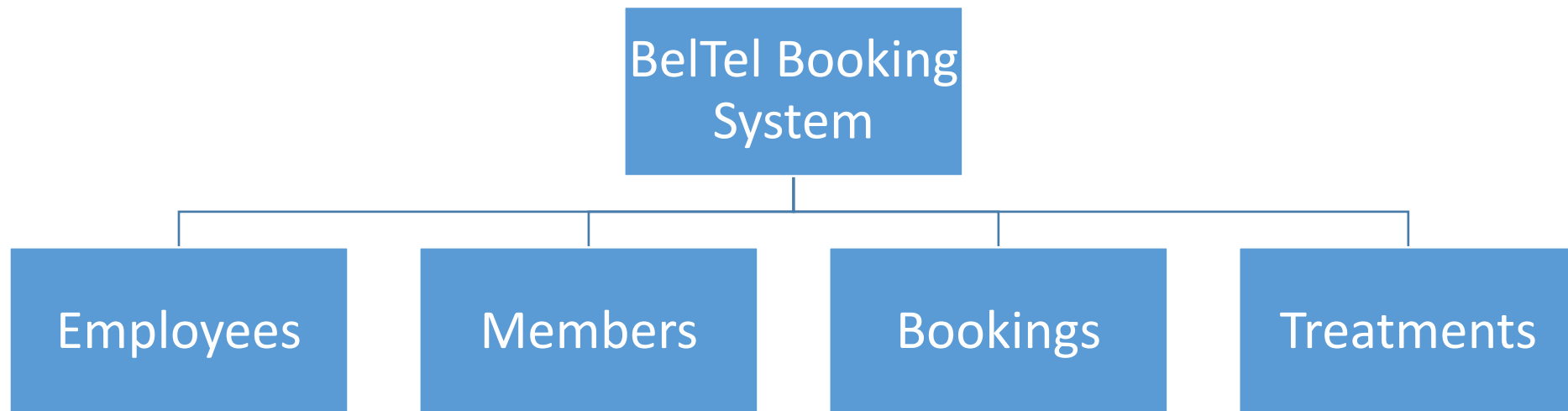
Purpose of Entity: To link tblMember to tblMedical to prevent a Many-to-Many relationship which would cause prevent data redundancy and also to store data relating to medical restrictions.

Entity Name:	tblMemberMedical	Primary Key:	MemberMedicalID	Foreign Key:	MedicalID	Related To:	tblMedical
--------------	------------------	--------------	-----------------	--------------	-----------	-------------	------------

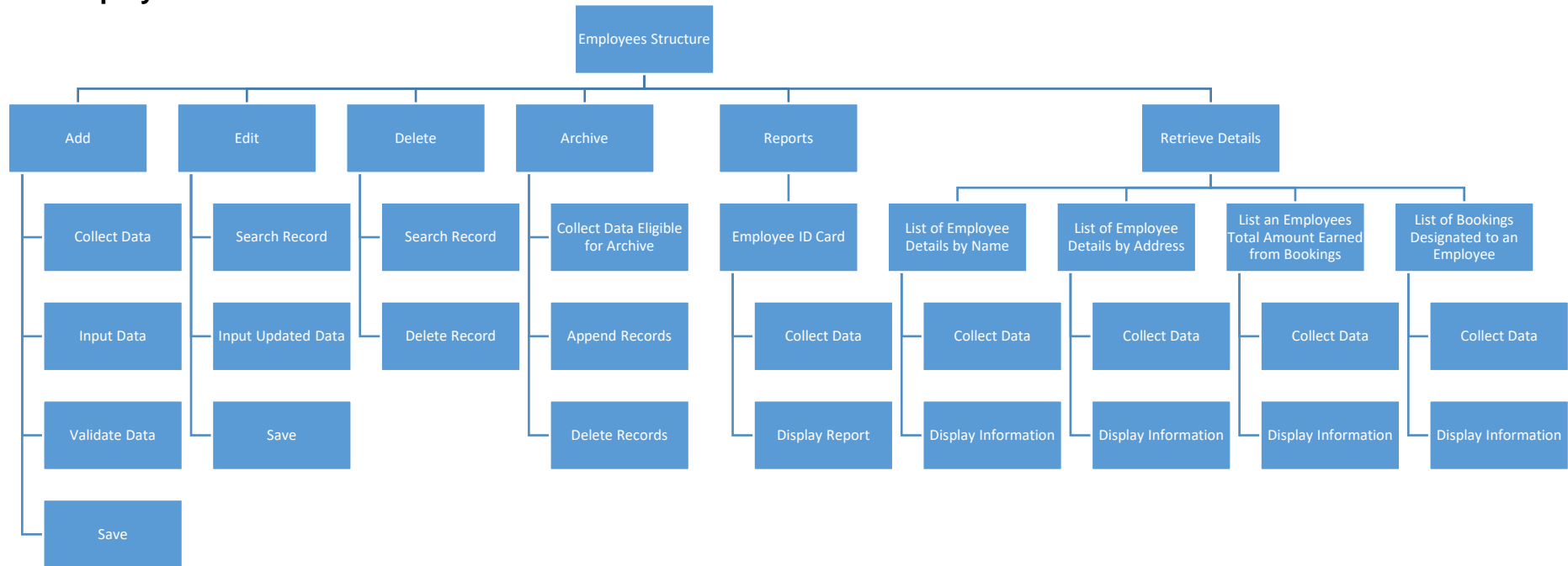
PK	Field Name	R	I	Data Type	Field Size	IM/ VR/ EM	Example Date
✓	MemberMedicalID	✓	X	AutoNumber			1
	MedicalID	✓	*	Number	ODP	VR: Lookup from tblMember	1
	Restrictions			Long Text			Allergic to nuts

2.4 Structure Diagram

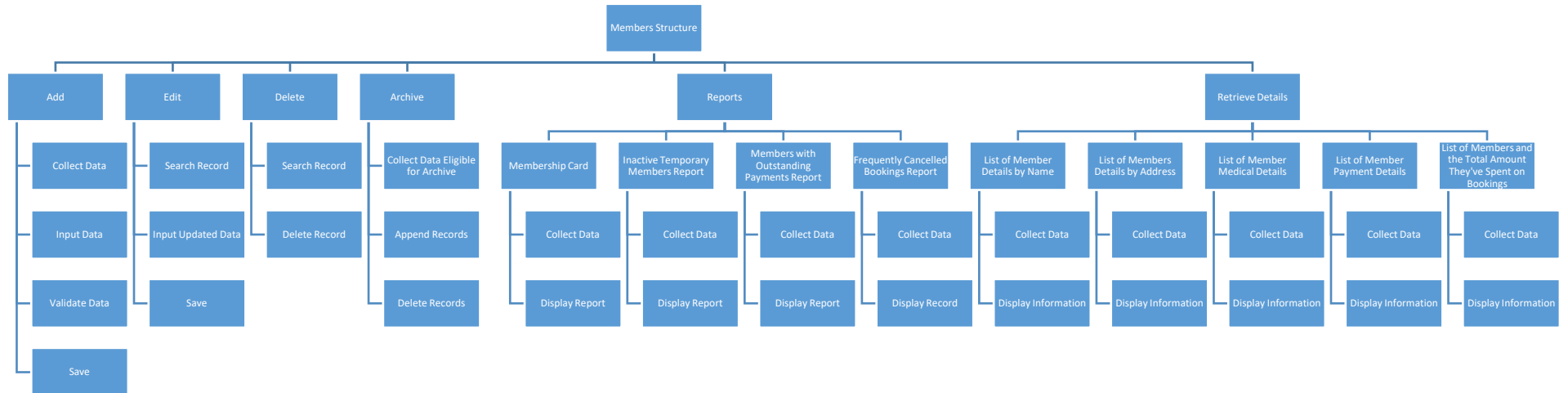
2.4.1 Overview Structure



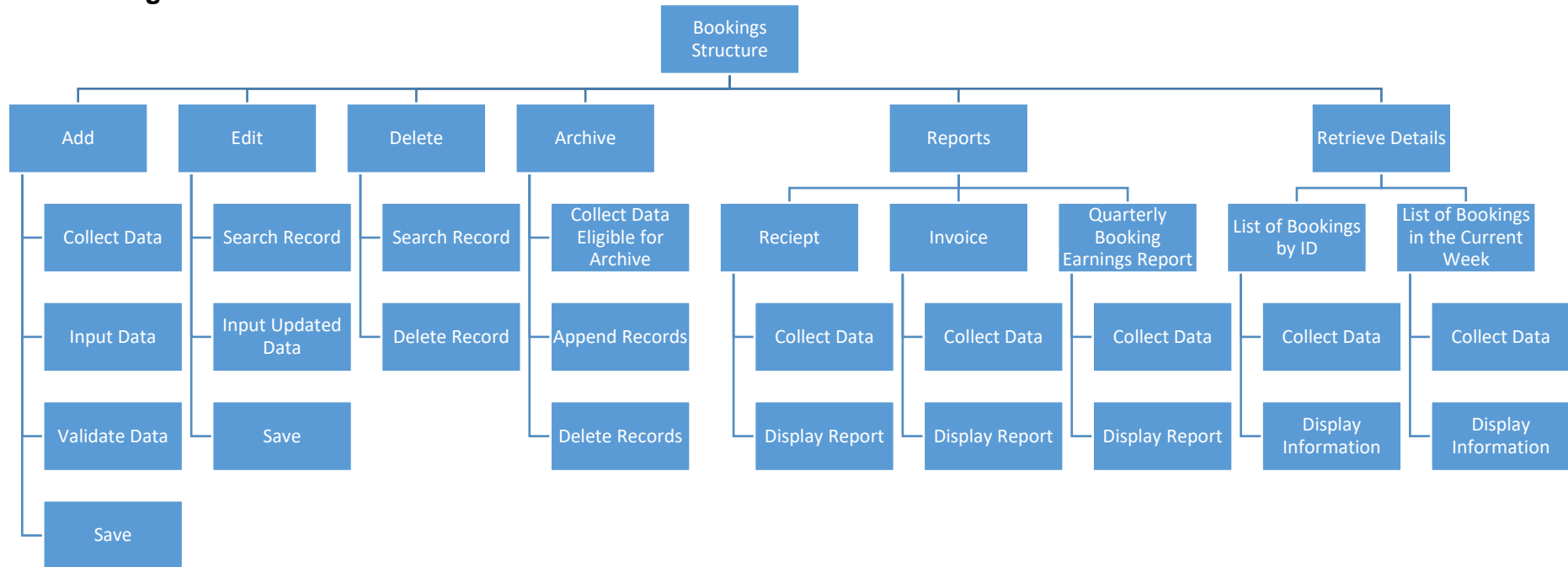
2.4.2 Employees Structure



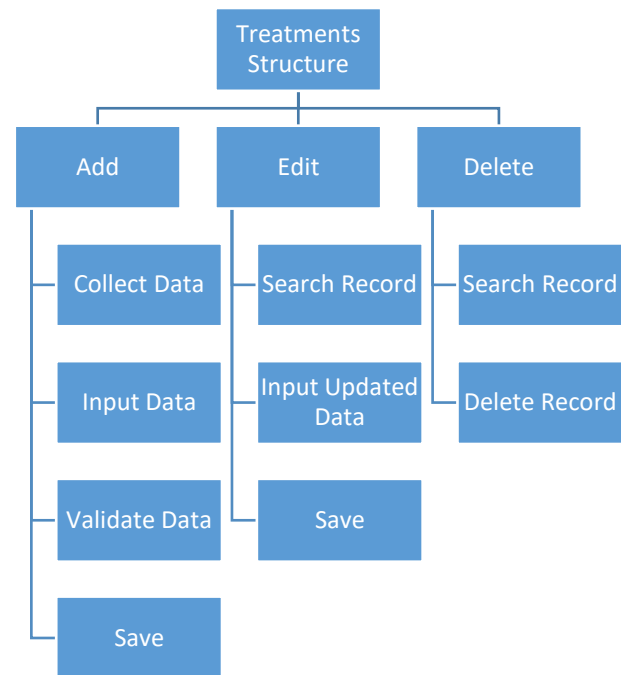
2.4.3 Members Structure



2.4.4 Bookings Structure



2.4.5 Treatments Structure



2.5 Form Design

Header Formatting:

Back Color: #DFDBE7

Title

Font Name: *Calibri (Detail)*
Font Size: *48*
Font Weight: *Bold*
Fore Color: *#604C78*
Border Color: *No Color*
Text Align: *General*

Logo

Name: *Beltel Logo Cropped*
Location: *D:\Year 14\Digital Technology\Mrs Nevin\Portfolio Data*
File Type: *PNG*
Width: *4.711cm*
Height: *2.981cm*

Box

Back Color: *#F3F0F6*
Border Style: *Transparent*

Line

Border Style: *Solid*
Border Width: *3 pt*
Border Color: *#6F3198*

Represented by:

General Command Buttons

Back Color: *#DFDBE7*
Border Style: *Solid*
Border Width: *Hairline*
Border Color: *#9F8CB7*
Hover Color: *#DFDBE7*
Pressed Color: *#DFDBE7*
Hover Fore Color: *#404040*
Pressed Fore Color: *#404040*
Alignment: *Center*

This applies to: *cmdSave, CmdRefresh, cmdFirst, cmdPrevious, cmdNext, cmdLast, cmdPrint, cmdBrowse*

These will be represented by the following in the same order:



Exit Command Button

Back Color: *No Color*
Hover Color: *#FFFFFFFE*
Pressed Color: *#FFFFFFFE*
Hover Fore Color: *#404040*
Pressed Fore Color: *#404040*

This applies to: *cmdExit*

This is represented by:



Body Formatting:

Back Color: #FFFFFF

Alternate Back Color: #F2F2F2

Headings

Font Name: *Calibri (Detail)*
Font Size: 18
Font Weight: **Bold**
Fore Color: #9F8CB7
Border Color: No Color
Text Align: General

Labels

Font Name: *Calibri (Detail)*
Font Size: 11
Font Weight: Normal
Fore Color: #BFB2CF
Border Color: No Color
Text Align: General

Text Box

Border Style: Solid
Border Width: Hairline
Border Color: #BFB2CF
Font Name: *Calibri (Detail)*
Font Size: 11
Font Weight: Normal
Fore Color: #222A35
Text Align: General

Footer Formatting:

Back Color: #F3F0F6

Line

Border Style: Solid
Border Width: 3 pt
Border Color: #6F3198

Represented by: _____

General Command Buttons

Back Color: #F3F0F6
Border Style: Solid
Border Width: Hairline
Border Color: #9F8CB7
Hover Color: # F3F0F6
Pressed Color: # F3F0F6
Hover Fore Color: #9F8CB7
Pressed Fore Color: #9F8CB7
Font Name: *Calibri (Detail)*
Font Size: 11
Font Weight: Normal
Fore Color: #9F8CB7
Alignment: Center

This applies to: *cmdAddRecord, cmdDeleteRecord*

Sub-Form Formatting:

Back Color: *#FFFFFF*

Text Command Buttons

Back Color: *#FFFFFF*
Border Style: *Solid*
Border Width: *Hairline*
Border Color: *#9F8CB7*
Hover Color: *# F3F0F6*
Pressed Color: *# F3F0F6*
Hover Fore Color: *#9F8CB7*
Pressed Fore Color: *#9F8CB7*
Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *#9F8CB7*
Alignment: *Center*

This applies to: *cmdAddRecord, cmdDeleteRecord*

Image Command Buttons

Back Color: *#FFFFFF*
Border Style: *Solid*
Border Width: *Hairline*
Border Color: *#9F8CB7*
Hover Color: *#DFDBE7*
Pressed Color: *#DFDBE7*
Hover Fore Color: *Text 1, Lighter 25%*
Pressed Fore Color: *Text 1, Lighter 25%*
Alignment: *Center*

This applies to: *cmdSave, CmdRefresh, cmdFirst, cmdPrevious, cmdNext, cmdLast*

Line (Doesn't apply to Headings)

Border Style: *Solid*
Border Width: *1 pt*
Border Color: *#BFB2CF*

Represented by: 

Key

L	Label	T	Title	H	Header	LG	Logo	TB	Text Box	CDB	Command Button
CB	Combo Box	OG	Option Group	L	Line	B	Box	CKB	Check Box		

Macros:

RunMacro	MacroName	qryUpdBookingEndTime
	Repeat Count	Datasheet
	Repeat Expression	Edit

Header:

Data Source: SELECT [tblMembers].[MemberID], [tblMembers].[Firstname], [tblMembers].[Surname] FROM tblMembers;

H: Booking Details											
L: Member Name		CB: SQL									
L: Treatments		CB: TreatmentName		L: Group Booking		CKB		L: Group Size		TB: GroupSize	
L: Employee Name		CB: SQL									
L: Date of Booking		TB: DateOfBooking		L: Time Start		TB: TimeStart		L: End Time		TB: EndTime	
OG: Booking Status											

Data Source:	SELECT	[tblEmployee].[EmployeeID],	[tblEmployee].[Firstname],	[tblEmployee].[Surname]	FROM	tblEmployee;
--------------	--------	-----------------------------	----------------------------	-------------------------	------	--------------

CDB: Add New Booking Record	CDB: Delete Booking Record
-----------------------------	----------------------------

2.5.2 Form 2: frmEmployee

Data Source: tblEmployee

Purpose: To add, edit, view and delete Employee records.

Algorithms:

The code that will be implemented into Form_Current will convert the text present in the database into a numeric value which can be displayed in an "Option Group", this is done each time it is used.

While the code that will be implented in frm*_AfterUpdate converts the numeric value outputted by the Option Group into text so it can be properly displayed in the database without breaching any validation rules or data types.

```
Private Sub Form_Current()  
Select Case Title.Value  
Case "Mr"  
frmTitle.Value = 1  
Case "Mrs"  
frmTitle.Value = 2  
Case "Miss"  
frmTitle.Value = 3  
Case "Dr"  
frmTitle.Value = 4  
Case "Rev"  
frmTitle.Value = 5  
End Select
```

```
Select Case Gender.Value  
Case "M"  
frmGender.Value = 1  
Case "F"  
frmGender.Value = 2  
End Select
```

```
Select Case Status.Value  
Case "Part-Time"  
frmStatus.Value = 1  
Case "Full-Time"  
frmStatus.Value = 2  
End Select
```

```
End Sub
```

```
Private Sub frmGender_AfterUpdate()  
Select Case frmGender.Value  
Case 1  
Gender = "M"  
Case 2  
Gender = "F"  
End Select  
End Sub
```

```
Private Sub frmStatus_AfterUpdate()  
Select Case frmStatus.Value  
Case 1  
Status = "Part-Time"  
Case 2  
Status = "Full-Time"  
End Select  
End Sub
```

```

Private Sub frmTitle_AfterUpdate()
Select Case frmTitle.Value
Case 1
Title = "Mr"
Case 2
Title = "Mrs"
Case 3
Title = "Miss"
Case 4
Title = "Dr"
Case 5
Title = "Rev"
End Select
End Sub

```

This code will enable you to search through various file directories by opening the File Dialog window, when an image is selected it will store the folder directory and file name as text in the database and will be displayed as an image.

```

Private Sub frmBrowse_Click()

Dim f As Object
Dim strfile As String
Dim strfolder As String
Dim VarItem As Variant

Set f = Application.FileDialog(3)
f.allowMultiSelect = True
If f.Show Then
For Each VarItem In f.selectedItems
strfile = Dir(VarItem)

strfolder = Left(VarItem, Len(VarItem) - Len(strfile))
MsgBox "Folder" & strfolder & vbCrLf & _
"File: " & strfile

Photograph_Directory = strfolder + strfile
Next
End If
Set f = Nothing

End Sub

```

Calculations: =DateDiff("mm",Date(),[DateOfEmployment])

Header:

Data Source: SELECT [tblEmployee].[EmployeeID], [tblEmployee].[Firstname], [tblEmployee].[Surname] FROM tblEmployee;									
LG		T: Employee Form							
		L: Employee ID:	TB: EmployeeID						L: Search: CB: SQL
<div style="text-align: center;">Box</div>									

Body:

Photograph Data Source: Photograph Directory	H: Booking Details		
	OG: Title		
	L: First Name	TB: FirstName	OG: Gender
	L: Surname	TB: Surname	
	L: Date of Birth	TB: DOB	
	H: Contact Details		
	L: Phone Number	TB: PhoneNumber	
	L: Email Address	TB: EmailAddress	
	L: Address	TB: Address	
	L: Town	TB: Town	
L: County	CB: County		
L: Postcode	TB: Postcode		
H: Contact Details			
L: Department			
CB: Department			
L: Position			
CB: Position			
L: National Insurance Number			
TB: NationalInsuranceNo			
L: Date Of Employment			
TB: DateOfEmployment			
L: Contracted Hours			
TB: ContractedHours			
L: File Path	TB: Photograph Directory		
OG: Status			
H: Contact Details			
TB: Biograph			
L: No. of Months Employed:	TB: =DateDiff("mm",Date(),[DateOfEmployment])	L: Archive	CKB

Footer:

CDB: Add New Employee Record	CDB: Delete Employee Record
------------------------------	-----------------------------

2.5.3 Form 3: frmMember

Data Source: tblMembers, tblMedical, tblMemberMedical, tblPayments, tblPaymentMethod, tblMemberMembership, tblMembershipLevel, tblSubscriptionPlan

Purpose: To add, edit, view and delete records relating to Members.

Code:

The code that will be implemented into Form_Current will convert the text present in the database into a numeric value which can be displayed in an "Option Group", this is done each time it is used.

While the code that will be implented in frm*_AfterUpdate converts the numeric value outputted by the Option Group into text so it can be properly displayed in the database without breaching any validation rules or data types.

```
Private Sub Form_Current()  
Select Case Title.Value  
Case "Mr"  
frmTitle.Value = 1  
Case "Mrs"  
frmTitle.Value = 2  
Case "Miss"  
frmTitle.Value = 3  
Case "Dr"  
frmTitle.Value = 4  
Case "Rev"  
frmTitle.Value = 5  
End Select  
  
Select Case Gender.Value  
Case "M"  
frmGender.Value = 1  
Case "F"  
frmGender.Value = 2  
End Select  
  
Select Case TypeOfMember.Value  
Case "Adult"  
frmTypeOfMember.Value = 1  
Case "Couple"  
frmTypeOfMember.Value = 2  
Case "Concessionary"  
frmTypeOfMember.Value = 3  
Case "Temporary"  
frmTypeOfMember.Value = 4  
End Select  
  
End Sub  
  
Private Sub frmGender_AfterUpdate()  
Select Case frmGender.Value  
Case 1  
Gender = "M"  
Case 2  
Gender = "F"  
End Select  
End Sub  
  
Private Sub frmTitle_AfterUpdate()  
Select Case frmTitle.Value  
Case 1  
Title = "Mr"
```

```

Case 2
Title = "Mrs"
Case 3
Title = "Miss"
Case 4
Title = "Dr"
Case 5
Title = "Rev"
End Select
End Sub

```

```

Private Sub frmTypeOfMember_AfterUpdate()
Select Case frmTypeOfMember.Value
Case 1
TypeOfMember = "Adult"
Case 2
TypeOfMember = "Couple"
Case 3
TypeOfMember = "Concessionary"
Case 4
TypeOfMember = "Temporary"
End Select
End Sub

```

This code will enable you to search through various file directories by opening the File Dialog window, when an image is selected it will store the folder directory and file name as text in the database and will be displayed as an image.

```

Private Sub frmBrowse_Click()

Dim f As Object
Dim strfile As String
Dim strfolder As String
Dim VarItem As Variant

Set f = Application.FileDialog(3)
f.allowMultiSelect = True
If f.Show Then
For Each VarItem In f.selectedItems
strfile = Dir(VarItem)

strfolder = Left(VarItem, Len(VarItem) - Len(strfile))
MsgBox "Folder" & strfolder & vbCrLf & _
"File: " & strfile

Photograph_Directory = strfolder + strfile
Next
End If
Set f = Nothing

End Sub

```

Calculations: =Count ([MedicalID]), =DateDiff("mm",Date(),[DateJoined])

Header:

Data Source: SELECT [tblMembers].[MemberID], [tblMembers].[Firstname], [tblMembers].[Surname] FROM tblMembers;

LG

T: Member Form

L: Member ID:

TB: MemberID

L: Search:

CB: SQL

Body:

Photograph Data Source: Photograph Directory	H: Booking Details			
	OG: Title			
	L: First Name	TB: FirstName	OG: Gender	
	L: Surname	TB: Surname		
	L: Date of Birth	TB: DOB		
	H: Contact Details			
	L: Phone Number	TB: PhoneNumber		
	L: Email Address	TB: EmailAddress		
	L: Address	TB: Address		
	L: Town	TB: Town		
L: County	CB: County			
L: Postcode	TB: Postcode			
H: Contact Details				
OG: TypeOfMember		L: Date Joined	TB: Date Joined	
Sub-Form: frmMember_MembershipSubform				
OG: MembershipLevel		OG: Subscription Plan		
L: Paid Date	TB: PaidDate			
L: Expiration Date	TB: ExpirationDate			
L: File Path	TB: Photograph Directory			

Sub-Form: frmMember_MedicalSubform				
H: Medical Details	L: (Optional)	L: Medical ID:	TB: MedicalID	
L: Condition Name	TB: ConditionName	L: No. of Records:	TB: =Count ([MedicalID])	
Sub-Form: frmMember_RestrictionsSubform				
L: Restrictions	TB: Restrictions			
L: ID:	TB: Member MedicalID			
	CDB: Add New Restriction			CDB: Delete Restriction
CDB: Add New Medical Record			CDB: Delete Medical Record	

Sub-Form: frmMember_PaymentSubform							
H: Payment Details		L: Payment Method ID:		TB: PaymentMethodID			
L: Payment Method	CB: PaymentMethod	L: Payment Type	CB: PaymentType	L: Status	TB: Status		
L: Card Number	TB: CardNumber	L: Expiration Year	TB: ExpirationYear	L: Expiration Month	TB: ExpirationMonth	L: Security Number	TB: SecurityNumber
CDB: Add New Payment Method Record				CDB: Delete Payment Method Record			
L: Member for:		TB: =DateDiff("mm",Date(),[DateJoined])			L: Months.		L: Archive
							CKB

Footer:

CDB: Add New Member Record









CDB: Delete Member Record

2.5.4 Form 4: frmTreatment

Data Source: tblTreatment

Purpose: To add, edit, view and delete Treatment records.

Header:

Data Source: SELECT [tblTreatment].[TreatmentID], [tblTreatment].[TreatmentName] FROM tblTreatment;													
LG		T: Treatment Form											
 		L: Treatment ID:		TB: TreatmentID		    		L: Search:		CB: SQL			
Box													

Body:

H: Treatment Details											
L: Treatment Name		TB: TreatmentName		L: Duration		TB: Duration		L: Cost		TB: Cost	
L: Description		TB: Description									

Footer:





CDB: Add New Treatment Record		CDB: Delete Treatment Record	
-------------------------------	--	------------------------------	--

2.5.5 Form 5: frmMembership

Data Source: tblTreatment

Purpose: To add, edit, view and delete Membership records.

Header:

Data Source: SELECT [tblMembershipLevel].[MembershipLevelID], [tblMembershipLevel].[MembershipLevel] FROM tblMembershipLevel;															
LG		T: Membership Level Form													
		L: Membership Level ID:		TB: MembershipLevelID						L: Search:		CB: SQL			
<div>Box</div>															

Body:

H: Membership Level Details							
L: Membership Level		TB: MembershipLevel		L: Cost		TB: Cost	
L: Description		TB: Description					

Footer:

CDB: Add New Membership Level Record	CDB: Delete Membership Level Record
--------------------------------------	-------------------------------------

2.6 Query Design

2.6.1 Query 1: qryAppArchiveBookings

Purpose: Archive all “completed” or “cancelled” booking records.

Frequency: Adhoc

Data Source: tblBookings

Type: Append

Destination Source: tblArchivedBookings

Field:	BookingID	MemberID	EmployeeID	TreatmentID	GroupBooking	GroupSize	DateOfBooking	TimeStart	EndTime	Status
Table:	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings
Append To:	BookingID	MemberID	EmployeeID	TreatmentID	GroupBooking	GroupSize	DateOfBooking	TimeStart	EndTime	Status
Criteria:										'Completed' or 'Cancelled'

2.6.2 Query 2: qryAppArchiveEmployee

Purpose: Archive all Employees records marked to be archived.

Frequency: Adhoc

Data Source: tblEmployee

Type: Append

Destination Source: tblArchivedEmployee

Field:	EmployeeID	Title	Firstname	Surname	Gender	DOB	PhoneNumber	EmailAddresses	Address	Town	County	Postcode	Department	Position	NationalInsuranceNo	DateOfEmployment	Status	ContractedHours	Wage	Biograph	Photograph Directory	Archive
Table:	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee
Append To:	EmployeeID	Title	Firstname	Surname	Gender	DOB	PhoneNumber	EmailAddresses	Address	Town	County	Postcode	Department	Position	NationalInsuranceNo	DateOfEmployment	Status	ContractedHours	Wage	Biograph	Photograph Directory	Archive
Criteria:																						Yes

2.6.3 Query 3: qryAppArchiveMedical

Purpose: Archive all Medical records if an associated Member has been marked to be archived.

Frequency: Adhoc

Data Source: tblMedical, tblMemberMedical, tblMembers

Type: Append

Destination Source: tblArchivedMedical

Field:	MedicalID	MemberID	ConditionName	Archive
Table:	tblMedical	tblMedical	tblMedical	tblMembers
Append To:	MedicalID	MemberID	ConditionName	
Criteria:				Yes

2.6.4 Query 4: qryAppArchiveMemberMedical

Purpose: Archive all Medical records if an associated Member has been marked to be archived.

Frequency: Adhoc

Data Source: tblMemberMedical, tblMedical

Type: Append

Destination Source: tblArchivedMemberMedical

Field:	MemberMedicalID	MedicalID	Restrictions	Archive
Table:	tblMemberMedical	tblMemberMedical	tblMemberMedical	tblMembers
Append To:	MemberMedicalID	MedicalID	Restrictions	
Criteria:				Yes

2.6.5 Query 5: qryAppArchiveMemberMembership

Purpose: Archive all Membership records if an associated Member has been marked to be archived.

Frequency: Adhoc

Data Source: tblMemberMembership, tblMembers

Type: Append

Destination Source: tblArchivedMemberMembership

Field:	MemberMembershipID	MemberID	MembershipID	SubscriptionPlanID	PaidDate	ExpirationDate	Archive
Table:	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMembers
Append To:	MemberMembershipID	MemberID	MembershipID	SubscriptionPlanID	PaidDate	ExpirationDate	
Criteria:							Yes

2.6.6 Query 6: qryAppArchiveMembers

Purpose: Archive all Members records marked to be archived.

Frequency: Adhoc

Data Source: tblMembers

Type: Append

Destination Source: tblArchivedMembers

Field:	MemberID	TypeOfMember	Title	Firstname	Surname	Gender	PhoneNumber	EmailAddress	Address	Town	County	Postcode	DateJoined	Loyalty	EligibleDiscount	DOB	Photograph Directory	Archive
Table:	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers

Append To:	MemberID	TypeOfMember	Title	Firstname	Surname	Gender	PhoneNumber	EmailAddress	Address	Town	County	Postcode	DateJoined	Loyalty	EligibleDiscount	DOB	Photograph Directory	Archive
Criteria:																		Yes

2.6.7 Query 7: qryAppArchivePaymentMethod

Purpose: Archive all Payment Method records if an associated Member has been marked to be archived.

Frequency: Adhoc

Data Source: tblPaymentMethod, tblPayments

Type: Append

Destination Source: tblArchivedPaymentMethod

Field:	PaymentMethodID	PaymentID	PaymentMethod	PaymentType	CardNumber	ExpirationMonth	ExpirationYear	SecurityNumber	Archive
Table:	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblMembers
Append To:	PaymentMethodID	PaymentID	PaymentMethod	PaymentType	CardNumber	ExpirationMonth	ExpirationYear	SecurityNumber	
Criteria:									Yes

2.6.8 Query 8: qryAppArchivePayments

Purpose: Archive all Payment records if an associated Member has been marked to be archived.

Frequency: Adhoc

Data Source: tblPayments, tblPaymentMethod, tblMembers

Type: Append

Destination Source: tblArchivedPayments

Field:	PaymentID	MemberID	AmountOutstanding	Status	Archive
Table:	tblPayments	tblPayments	tblPayments	tblPayments	tblMembers
Append To:	PaymentID	MemberID	AmountOutstanding	Status	
Criteria:					Yes

2.6.9 Query 9: qryDelArchivedBookings

Purpose: Delete all Booking records that have already been archived.

Frequency: Adhoc

Data Source: tblBookings

Type: Delete

Field:	BookingID	MemberID	EmployeeID	TreatmentID	GroupBooking	GroupSize	DateOfBooking	TimeStart	EndTime	Status
Table:	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings

Delete:	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where
Criteria:										'Completed' or 'Cancelled'

2.6.10 Query 10: qryDelArchivedEmployee

Purpose: Delete all Employee records that have already been archived.

Frequency: Adhoc

Data Source: tblEmployee

Type: Delete

Field:	EmployeeID	Title	Firstname	Surname	Gender	DOB	PhoneNumbers	EmailAddresses	Address	Town	County	Postcode	Department	Position	NationalInsuranceNo	DateOfEmployment	Status	ContractedHours	Wage	Biograph	PhotographDirectory	Archive
Table:	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee
Delete:	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where
Criteria:																						Yes

2.6.11 Query 11: qryDelArchivedMedical

Purpose: Delete all Medical records that have already been archived.

Frequency: Adhoc

Data Source: tblMedical, tblMemberMedical, tblMembers

Type: Delete

Field:	MedicalID	MemberID	ConditionName	Archive
Table:	tblMedical	tblMedical	tblMedical	tblMembers
Delete:	Where	Where	Where	
Criteria:				Yes

2.6.12 Query 12: qryDelArchivedMemberMedical

Purpose: Delete all Medical records that have already been archived.

Frequency: Adhoc

Data Source: tblMemberMedical, tblMedical

Type: Delete

Field:	MemberMedicalID	MedicalID	Restrictions	Archive
Table:	tblMemberMedical	tblMemberMedical	tblMemberMedical	tblMembers
Delete:	Where	Where	Where	
Criteria:				Yes

2.6.13 Query 13: qryDelArchivedMemberMembership

Purpose: Delete all Members' Membership records that have already been archived.

Frequency: Adhoc

Data Source: tblMemberMembership, tblMembers

Type: Delete

Field:	MemberMembershipID	MemberID	MembershipID	SubscriptionPlanID	PaidDate	ExpirationDate	Archive
Table:	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMemberMembership	tblMembers
Delete:	Where	Where	Where	Where	Where	Where	
Criteria:							Yes

2.6.14 Query 14: qryDelArchivedMembers

Purpose: Delete all Member records that have already been archived.

Frequency: Adhoc

Data Source: tblMembers

Type: Delete

Field:	MemberID	TypeOfMember	Title	Firstname	Surname	Gender	PhoneNumber	EmailAddress	Address	Town	County	Postcode	DateJoined	Loyalty	EligibleDiscount	DOB	Photograph Directory	Archive
Table:	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers
Delete:	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where	Where
Criteria:																		Yes

2.6.15 Query 15: qryDelArchivedPaymentMethod

Purpose: Delete all Payment Method records that have already been archived.

Frequency: Adhoc

Data Source: tblPaymentMethod, tblPayments

Type: Delete

Field:	PaymentMethodID	PaymentID	PaymentMethod	PaymentType	CardNumber	ExpirationMonth	ExpirationYear	SecurityNumber	Archive
Table:	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblMembers
Delete:	Where	Where	Where	Where	Where	Where	Where	Where	
Criteria:									Yes

2.6.16 Query 16: qryDelArchivedPayments

Purpose: Delete all Payment records that have already been archived.

Frequency: Adhoc

Data Source: tblPayments, tblPaymentMethod, tblMembers

Type: Delete

Field:	PaymentID	MemberID	AmountOutstanding	Status	Archive
Table:	tblPayments	tblPayments	tblPayments	tblPayments	tblMembers
Delete:	Where	Where	Where	Where	
Criteria:					Yes

2.6.17 Query 17: qryBookingLookup

Purpose: Lookup Booking details from the BookingID.

Frequency: Adhoc

Data Source: tblBookings, tblEmployee, tblMembers, tblTreatments

Type: Select

Additional Information:

Full Name and Assigned Employee are concatenations of Firstname and Surname fields in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are “*Sub-Total*” which is the result of the TreatmentCost multiplied by GroupSize, this is done in order to find the total cost of each person allocated in the booking, further more “*Eligible Discount*” has been added which multiplies the Sub-Total by Eligible Discount which will calculate the amount that can be discounted; Finally, “*Total*” has been added which is Sub-Total subtracted by Eligible Discount to find the final price the booking will cost.

The Criteria has been used effectively to ensure you can find a Booking by its ID, that the Assigned Employee field is not left blank as an employee is required to preform the treatments, that the time start and time end fall between 09:00 and 22:00 as that is BelTels opening hours and that the Status field is not “Cancelled” as the booking information will no longer be relevant.

Field:	BookingID	Full Name: [tblMembers][Firstname]+[tblMembers][Surname]	Phone Number: [tblMembers][PhoneNumber]	Treatment: TreatmentName	Initial Treatment Cost: Cost	Treatment Duration: Duration	Assigned Employee: [tblEmployee][Firstname]+[tblEmployee][Surname]	Group Booking: GroupBooking	Size Of Group: GroupSize	Date Of Booking: DateOfBooking	Time Start: TimeStart	Time End: EndTime	Sub-Total [tblTreatments][Cost]*[tblBookings][GroupSize]	Eligible Discount: Cost*[tblMembers][EligibleDiscount]	[Sub-Total Total Cost]-[Eligible Discount]	Status
Table:	tblBookings			tblTreatments	tblTreatment	tblTreatment		tblBookings	tblBookings	tblBookings	tblBookings	tblBookings				tblBookings
Sort:																
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:	[Enter Booking ID]						Is Not Null				>#09:00:00 #	<=#22:00:00 #				Not Like "Cancelled"

2.6.18 Query 18: qryBookingQuarterlyEarnings

Purpose: View earnings for any given quarter.

Frequency: Adhoc

Data Source: tblMembers, tblBooking, qryBookingLookup

Type: Select

Additional Information:

Full Name and Assigned Employee are concatenations of Firstname and Surname fields in order to reduce the amount of rows utilized and to group information in a manner that is professional.

The Criteria has been used effectively to ensure that the Date Of Booking falls within the current quarter and that the Status field is not “Cancelled” as the booking information will no longer be relevant.

Field:	Member ID: MemberID	Full Name: [tblMembers]![Firstname]+*+[tblMembers]![Surname]	Phone Number: [tblMembers]![PhoneNumber]	Date Of Booking: DateOfBooking	Total	Status
Table:	tblMembers			tblBookings	qryBookingLookup	tblBookings
Total:	Group By	Group By	Group By	Group By	Sum	Group By
Sort:						
Show:	✓	✓	✓	✓	✓	✓
Criteria:				DatePart("qq",Date())		Not Like "Cancelled"

2.6.19 Query 19: qryBookingsWeekly

Purpose: View Bookings for a given week.

Frequency: Weekly

Data Source: tblBookings, tblEmployee, tblMembers, tblTreatments

Type: Select

Additional Information:

Full Name and Assigned Employee are concatenations of Firstname and Surname fields in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are “*Sub-Total*” which is the result of the TreatmentCost multiplied by GroupSize, this is done in order to find the total cost of each person allocated in the booking, further more “*Eligible Discount*” has been added which multiplies the Sub-Total by Eligible Discount which will calculate the amount that can be discounted; Finally, “*Total*” has been added which is Sub-Total subtracted by Eligible Discount to find the final price the booking will cost.

The Criteria has been used effectively to ensure the Assigned Employee field is not left blank as an employee is required to preform the treatments, that the Date Of Booking falls within the current week, that the time start and time end fall between 09:00 and 22:00 as that is BelTels opening hours and that the Status field is not “Cancelled” as the booking information will no longer be relevant.

Field:	BookingID	Full Name: [tblMembers]![Firstname]+*+[tblMembers]![Surname]	Phone Number: [tblMembers]![PhoneNumber]	Treatment: TreatmentName	Initial Treatment Cost: Cost	Treatment Duration: Duration	Assigned Employee: [tblEmployee]![Firstname]+*+[tblEmployee]![Surname]	Group Booking: GroupBooking	Size Of Group: GroupSize	Date Of Booking: DateOfBooking	Time Start: TimeStart	Time End: TimeEnd	Sub-Total [tblTreatments]![Cost]*[tblBookings]![GroupSize]	Eligible Discount: [tblMembers]![EligibleDiscount]	[Sub-Total Cost]-[Eligible Discount]	Total [Sub-Total Cost]-[Eligible Discount]	Status
Table:	tblBookings			tblTreatments	tblTreatments	tblTreatments		tblBookings	tblBookings	tblBookings	tblBookings	tblBookings					tblBookings
Sort:																	
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Criteria							Is Not Null			DatePart("ww",Date())	>#09:00:00 #	<=#22:00:00 #						Not Cancelled	Like "
----------	--	--	--	--	--	--	-------------	--	--	---------------------------	-----------------	------------------	--	--	--	--	--	------------------	-----------

2.6.20 Query 20: qryEmployeeAddressLookup

Purpose: Lookup Employee details from their address

Frequency: Adhoc

Data Source: tblEmployee

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are "*Salary*" which multiplies the Wage by ContractedHours by 52 in order to calculate the total amount an employee would earn on that contract over the period of a year.

The Criteria has been used effectively to ensure you can look up their address or any address containing key words specified, that the Employee is of age to work (in the UK that is 16), that their Date of Employment is before now as you cannot be employed in the future, that their contract is greater than or equal to 8 hours since BelTel only offer 8 hour contracts to their employees and that Archive is not null so the employee being searched for is still employed by BelTel

Field:	Employee ID	Title	Full Name: [tblEmployee][Firstname] & [tblEmployee][Surname]	Gender	Date Of Birth: DOB	Phone Number: PhoneNumb er	Email Address: EmailAddress	Address: [tblEmployee][Address] & [tblEmployee][Town] & [tblEmployee][County] & [tblEmployee][Postcod e]	Departmen t	Position	National Insurance Number: NationalInsurance No	Date Of Employment: DateOfEmployment	Status	Contracted Hours: ContractedHours	Wage	Salary: [Wage]*[ContractedHours]* 52	Biograph	Photograph	Archive
Table:	tblEmployee	tblEmployee		tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee		tblEmployee	tblEmployee	tblEmployee
Sort:																			
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria					<=DateAdd("yyyy" ,-16,Date())			Like "*" & [Enter Address] & "*"				<=Now()		>=8					Is Not Null

2.6.21 Query 21: qryEmployeeBookingEarnings

Purpose: Lookup Bookings designated to Employees by their name.

Frequency: Adhoc

Data Source: tblEmployee, tblBookings, qryBookingLookup

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are "*Salary*" which multiplies the Wage by ContractedHours by 52 in order to calculate the total amount an employee would earn on that contract over the period of a year.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified and that their contract is greater than or equal to 8 hours since BelTel only offer 8 hour contracts to their employees.

Field:	EmployeeID	Title	Full Name: [tblEmployee][Firstname]+ "[tblEmployee][Surname]"	Department	Position	Status	Contracted Hours: ContractedHours	Wage	Salary: [Wage]*[ContractedHours]*52	Total
Table:	tblEmployee	tblEmployee		tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee		qryBookingLookup
Total:	Group By	Group By	Group By	Group By	Group By	Group By	Group By	Group By	Group By	Sum
Sort:										
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:			Like "*" & [Enter Employee Name] & "*"				>=8			

2.6.22 Query 22: qryEmployeeBookings

Purpose: Lookup Bookings designated to Employees by their name.

Frequency: Adhoc

Data Source: tblEmployee, tblBookings, tblTreatments

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified, that the Date Of Booking is on the current date, that the time start and time end fall between 09:00 and 22:00 as that is BelTels opening hours and that the Status field is not "Cancelled" as the booking information will no longer be relevant.

Field:	EmployeeID	Full Name: [tblEmployee][Firstname]+ "[tblEmployee][Surname]"	Date Of Booking: DateOfBooking	Treatment: TreatmentName	Duration	Group Booking: GroupBooking	Size Of Group: GroupSize	Time Start: TimeStart	Time End: EndTime	Status
Table:	tblEmployee		tblBookings	tblTreatments	tblTreatments	tblBookings	tblBookings	tblBookings	tblBookings	tblBookings
Sort:										
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:		Like "*" & [Enter Employee Name] & "*"	>=Now()					>#09:00:00#	<=#22:00:00#	Not Like "Cancelled"

2.6.22 Query 22: qryEmployeeLookup

Purpose: Lookup Employee details by their name.

Frequency: Adhoc

Data Source: tblEmployee

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are “*Salary*” which multiplies the Wage by ContractedHours by 52 in order to calculate the total amount an employee would earn on that contract over the period of a year.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified, that the Employee is of age to work (in the UK that is 16), that their Date of Employment is before now as you cannot be employed in the future, that their contract is greater than or equal to 8 hours since BelTel only offer 8 hour contracts to their employees and that Archive is not null so the employee being searched for is still employed by BelTel

Field:	EmployeeID	Title	Full Name: [tblEmployee].[FirstName] & [tblEmployee].[Surname]	Gender	Date Of Birth: DOB	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblEmployee].[Address] & [tblEmployee].[Town] & [tblEmployee].[County] & [tblEmployee].[Postcode]	Department	Position	National Insurance Number: NationalInsuranceNo	Date Of Employment: DateOfEmployment	Status	Contracted Hours: ContractedHours	Wage	Salary: [Wage]*[ContractedHours]*52	Biograph	Photograph	Archive
Table:	tblEmployee	tblEmployee		tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee	tblEmployee		tblEmployee	tblEmployee	tblEmployee
Sort:																			
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:			Like *** & [Enter Employee Name] & ***		<=DateAdd("yyyy",-16,Date())							<=Now()		>=8					Is Not Null

2.6.23 Query 23: qryMemberAddressLookup

Purpose: Lookup Member details by their address.

Frequency: Adhoc

Data Source: tblMembers, tblMembershipLevel, tblMemberMembership, tblSubscriptionPlan

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are “*Cost Per Transaction*” which multiplies the Membership Level Cost by the Subscription Plan Cost Multiplier in order to find how much the user will be paying for their membership fees depending on if its quarterly, monthly or yearly and their membership level selected.

The Criteria has been used effectively to ensure you can look up their address or any address containing key words specified, that the Member is at least 18 in order to utilize BelTels services as stated and that Archive is not null so the member still has a membership with BelTel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers].[FirstName] & [tblMembers].[Surname]	Gender	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblMembers].[Address] & [tblMembers].[Town] & [tblMembers].[County] & [tblMembers].[Postcode]	Date Of Birth: DOB	Membership Level: MembershipLevel	Subscription Plan: SubscriptionPlan	Cost Per Transaction: [tblMembershipLevel].[Cost]*[tblSubscriptionPlan].[CostMultiplier]	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers	tblMembers		tblMembers	tblMembershipLevel	tblSubscriptionPlan		
Sort:	Ascending											
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Criteria:							Like *** & [Enter Address] & ***	[Enter <=DateAdd("yyyy",-18,Date())					Is Not Null
-----------	--	--	--	--	--	--	----------------------------------	-------------------------------------	--	--	--	--	-------------

2.6.24 Query 24: qryMemberMedicalRecords

Purpose: Lookup Member Medical details by their name.

Frequency: Adhoc

Data Source: tblMedical, tblMemberMedical, tblMembers

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified, that you can look up their specific medical conditions in relation to any key words and that Archive is not null so the member still has a membership with BelTel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers][Firstname]+ "[tblMembers][Surname]"	Gender	Phone Number: PhoneNumber	Condition: ConditionName	Restrictions	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers	tblMedical	tblMemberMedical	tblMembers
Sort:	Ascending							
Show:	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:			*** & [Enter Customer Name] & ***			Like *** & [Enter Condition Name] & ***		Is Not Null

2.6.25 Query 25: qryMemberPaymentLookup

Purpose: Lookup Members' payment information by their name.

Frequency: Adhoc

Data Source: tblMembers, tblPayments, tblPaymentMethod

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified, that the Card Number field is not left blank or this information would be useless and that the Expiration Year is greater than or equal to now so its not expired.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers][Firstname]+ "[tblMembers][Surname]"	PaymentMethodID	Payment Method: PaymentMethod	Payment Type: PaymentType	Card Number: CardNumber	Expiration Month: ExpirationMonth	Expiration Year: ExpirationYear	Security Number: SecurityNumber	Status
Table:	tblMembers	tblMembers		tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPaymentMethod	tblPayments
Sort:	Ascending										
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:			*** & [Enter Customer Name] & ***				Is Not Null		>=Now()		
Or:											

2.6.26 Query 26: qryMembersBookingSpent

Purpose: View how much a given member has spent on bookings.

Frequency: Adhoc

Data Source: tblMembers, tblBookings, qryBookingLookup

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified and that the date joined is before today's date.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers]![Firstname]+ "[tblMembers]![Surname]	Date Joined: DateJoined	Loyalty	Eligible Discount: EligibleDiscount	Total
Table:	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	tblMembers	qryBookingLookup
Total:	Group By	Group By	Group By	Group By	Group By	Group By	Sum
Sort:	Ascending						
Show:	✓	✓	✓	✓	✓	✓	✓
Criteria:			"" & [Enter Customer Name] & ""	<=Now()			

2.6.27 Query 27: qryMembersLookup

Purpose: Lookup Member details by their name.

Frequency: Adhoc

Data Source: tblMembers, tblMembershipLevel, tblMemberMembership, tblSubscriptionPlan

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are “*Cost Per Transaction*” which multiplies the Membership Level Cost by the Subscription Plan Cost Multiplier in order to find how much the user will be paying for their membership fees depending on if its quarterly, monthly or yearly and their membership level selected.

The Criteria has been used effectively to ensure you can look up their name or any name containing key words specified, that the Member is at least 18 in order to utilize BelTels services as stated and that Archive is not null so the member still has a membership with BelTel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers]![Firstname]+ "[tblMembers]![Surname]	Gender	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblMembers]![Address] & " " & [tblMembers]![Town] & " " & [tblMembers]![County] & " " & [tblMembers]![Postcode]	Date Of Birth: DOB	Membership Level: MembershipLevel	Subscription Plan: SubscriptionPlan	Cost Per Transaction: [tblMembershipLevel]!Cost* [tblSubscriptionPlan]!CostMultiplier	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers	tblMembers		tblMembers	tblMembershipLevel	tblSubscriptionPlan		
Sort:	Ascending											
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:			"" & [Enter Customer Name] & ""					<=DateAdd("yyyy",-18,Date())				Is Not Null

2.6.28 Query 28: qryNoBookingsPerMember

Purpose: View how many bookings a single member has.

Frequency: Adhoc

Data Source: tblBookings, tblMembers

Type: Select

Additional Information: N/A

Field:	MemberID	BookingID
Table:	tblMembers	tblBookings
Total:	Group By	Count
Sort:		Descending
Show:	✓	✓
Criteria:		

2.6.29 Query 29: qryUpdBookingEndTime

Purpose: Updates the booking end time by adding the duration to the start time.

Frequency: Adhoc

Data Source: tblBookings, tblTreatments

Type: Update

Additional Information:

Calculating the sum of TimeStart and Duration will allow the EndTime to be presented.

Field:	EndTime
Table:	tblBookings
Update To:	[tblBookings].[TimeStart]+[tblTreatments].[Duration]
Criteria:	

2.6.30 Query 30: qryUpdCompleteBooking

Purpose: Updates the Booking Status to either "Completed" or "Cancelled", the choice of which being up to the employee

Frequency: Adhoc

Data Source: tblBookings

Type: Update

Additional Information:

The criteria allows the user to input the BookingID in order to find the record, following this if the BookingID enter has a Status of Booked they will be able to enter in information

Field:	Status	BookingID
--------	--------	-----------

Table:	tblBookings	tblBookings
Update To:	[Please enter new booking status]	
Criteria:	"Booked"	[Enter Booking ID]

2.6.31 Query 31: qryUpdExpiredMembers

Purpose: Update Members' Archive status to yes after their membership has expired.

Frequency: Monthly

Data Source: tblMembers, tblMemberMembership

Type: Update

Additional Information:

A Calculation has been added to figure out how many years has passed since their membership expired, if it becomes a year or greater their records will be archived.

Field:	Archive	DateDiff("yyyy",[ExpirationDate],Date())
Table:	tblMembers	
Update To:	Yes	
Criteria:	No	>=1

2.6.32 Query 32: qryUpdLoyaltyPoints

Purpose: Update the number of Member Loyalties they have from the amount of bookings they've made.

Frequency: Adhoc

Data Source: tblMembers, tblBookings

Type: Update

Additional Information: N/A

Field:	Loyalty	MemberID	Status
Table:	tblMembers	tblMembers	tblBookings
Update To:	[Loyalty]+1		
Criteria:		[frmBooking]![MemberID]	"Completed"

2.6.33 Query 33: qryUpdTemporaryMembers

Purpose: Update all Temporary Members' Archive status after 28 days from their join date to yes.

Frequency: Adhoc

Data Source: tblMembers

Type: Update

Additional Information:

Criteria has been implemented so that if a Member has a Member Type of Temporary add has exceeded more than 28 days since their join date then they will be archived.

Field:	Archive	TypeOfMember	DateJoined
Table:	tblMembers	tblMembers	tblMembers
Update To:	Yes		
Criteria:	No	"Temporary"	>DateAdd("d",28,Date())

2.6.34 Query 34: qryFrequentlyCancelledBookings

Purpose: Numbers the amount of cancellation members has incase BelTel needs to penalise them.

Frequency: Adhoc

Data Source: tblMembers, tblBooking, tblMembershipLevel, tblSubscriptionPlan

Type: Select

Additional Information:

This query will calculate the total number of bookings cancelled by members in order to determine users who may need a cooldown or restricted.

The Criteria has been used effectively to ensure that the Member is atleast 18 in order to utilize BelTels services as stated, the total number of of cancelled bookings is greater than or equal to free in order to make sure that the cancellations are actual frequent and that Archive is not null so the member still has a membership with BelTel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers].[Firstname]+ " " + [tblMembers].[Surname]	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblMembers].[Address] & " " & [tblMembers].[Town] & " " & [tblMembers].[County] & " " & [tblMembers].[Postcode]	Date Of Birth: DOB	Membership Level: MembershipLevel	Subscription Plan: SubscriptionPlan	No. Of Cancelled Bookings: Status	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers		tblMembers	tblMembershipLevel	tblSubscriptionPlan	tblBooking	tblMembers
Total:	Group By	Group By	Group By	Group By	Group By	Group By	Group By	Group By	Group By	Count	Group By
Sort:										Descending	
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:							<=DateAdd("yyyy",-18,Date())			>=3 AND "Cancelled"	Is Not Null

2.6.35 Query 35: qryInactiveTemporaryMembers

Purpose: List members if they exceed the temporary trial period in order to establish contact with them in an attempt to renew their membership.

Frequency: Adhoc

Data Source: tblMembers, tblMembershipLevel, tblMemberMembership, tblSubscriptionPlan

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are "Days Since Joined" Which finds the difference from the Members date joined and todays date, if it is 28 days or greater then it will be flagged by the criteria since members can only have a trial period of 28 days.

The Criteria has been used effectively to ensure that only Temporary members are flagged, that they have to exceed or be equal to 28 days in order to be deemed as inactive, that the Member is atleast 18 in order to utilize BelTels services as stated and that Archive is not null so the member still has a membership with BelTel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers]![Firstname]+ *[tblMembers]![Surname]	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblMembers]![Address] & " " & [tblMembers]![Town] & " " & [tblMembers]![County] & " " & [tblMembers]![Postcode]	Date Of Birth: DOB	Membership Level: MembershipLevel	Subscription Plan: SubscriptionPlan	Days Since Joined: =DateDiff("d",Date(),[DateJoined])	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers		tblMembers	tblMembershipLevel	tblSubscriptionPlan		
Sort:	Ascending										
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:		"Temporary"					<=DateAdd("yyyy",-18,Date())			>=28	Is Not Null

2.6.36 Query 36: qryOutstandingPayments

Purpose: Lists all members who have outstanding payments so that BelTel can use their contact information to retrieve what is owed.

Frequency: Adhoc

Data Source: tblMembers, tblMembershipLevel, tblMemberMembership, tblSubscriptionPlan, tblPayments

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

The Criteria has been used effectively to ensure that the amount outstanding is greater than zero and that Archive is not null so the member still has a membership with BelTel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers]![Firstname]+ *[tblMembers]![Surname]	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblMembers]![Address] & " " & [tblMembers]![Town] & " " & [tblMembers]![County] & " " & [tblMembers]![Postcode]	Membership Level: MembershipLevel	Subscription Plan: SubscriptionPlan	Amount Outstanding: AmountOutstanding	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers		tblMembershipLevel	tblSubscriptionPlan	tblPayments	
Sort:	Ascending									
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:									>0	Is Not Null

2.6.27 Query 27: qryInactiveMembers

Purpose: Lists all Inactive Members, this is required by BelTel so they can get in touch with past Members to convince them to renew their membership.

Frequency: Adhoc

Data Source: tblMembers, tblMembershipLevel, tblMemberMembership, tblSubscriptionPlan

Type: Select

Additional Information:

Full Name is a concatenation of Firstname and Surname while Address is a concatenation of Address, Town, County and Postcode fields; this is done in order to reduce the amount of rows utilized and to group information in a manner that is professional.

Calculations included are "Months since Membership Expired", this finds the difference between now and the expiration date in months so BelTel can determine if the Member is inactive.

The Criteria has been used effectively to ensure that atleast 12 months has past since the Member was active so they can now be deemed as inactive and that Archive is not null so the member still has a membership with BeITel at the time of search.

Field:	MemberID	Type Of Member: TypeOfMember	Full Name: [tblMembers]![Firstname]+ "+[tblMembers]![Surname]	Phone Number: PhoneNumber	Email Address: EmailAddress	Address: [tblMembers]![Address] & ", " & [tblMembers]![Town] & ", " & [tblMembers]![County] & ", " & [tblMembers]![Postcode]	Membership Level: MembershipLevel	Subscription Plan: SubscriptionPlan	Months since Membership Expired: =DateDiff("mm",Date(),[ExpirationDate])	Archive
Table:	tblMembers	tblMembers		tblMembers	tblMembers		tblMembershipLevel	tblSubscriptionPlan		
Sort:	Ascending									
Show:	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Criteria:									>=12	Is Not Null

2.7 Report Design

Report Header Formatting:

Back Color: #DFDBE7

Title

Font Name: Calibri (Detail)
Font Size: 48
Font Weight: Bold
Fore Color: #604C78
Border Color: No Color
Text Align: General

Logo

Name: Beltel Logo Cropped
Location: D:\Year 14\Digital Technology\Mrs Nevin\Portfolio Data
File Type: PNG
Width: 5.707cm
Height: 2.203cm

Command Buttons

Back Color: No Color
Hover Color: #FFFFFFFE
Pressed Color: #FFFFFFFE
Hover Fore Color: #404040
Pressed Fore Color: #404040

This applies to: *cmdExit, cmdPrint*

This is represented by:



Page Header Formatting:

Back Color: #F3F0F6

Line

Border Style: Solid
Border Width: 3 pt
Border Color: #6F3198

Represented by:



Labels

Font Name: Calibri (Detail)
Font Size: 11
Font Weight: Normal
Fore Color: #604C78
Border Color: No Color
Text Align: Left

Grouping Header:

Back Color: #FFFFFF

Alternate Back Color: #F9F7FB

Labels

Font Name: Calibri (Detail)
Font Size: 11

Font Weight: *Bold*
Fore Color: *#604C78*
Border Color: *No Color*
Text Align: *Left*

Detail Formatting:

Back Color: *#FFFFFF*
Alternate Back Color: *#F9F7FB*

Text Box

Border Style: *Solid*
Border Width: *Hairline*
Border Color: *#BFB2CF*
Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *#222A35*
Text Align: *General*

Page Footer Formatting:

Back Color: *#F3F0F6*

Line

Border Style: *Solid*
Border Width: *3 pt*
Border Color: *#6F3198*

Represented by: 

Text Box

Border Style: *Transparent*
Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *#604C78*
Text Align: *General*

Report Footer Formatting:

Back Color: *#FFFFFF*

Labels

Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Bold*
Fore Color: *#604C78*
Border Color: *No Color*
Text Align: *Left*

Text Box

Border Style: *Transparent*
Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *#604C78*
Text Align: *General*

Key

L	Label	T	Title	H	Header	LG	Logo	TB	Text Box	CDB	Command Button
----------	-------	----------	-------	----------	--------	-----------	------	-----------	----------	------------	----------------

CB	Combo Box	OG	Option Group	L	Line	B	Box	CKB	Check Box		
----	-----------	----	--------------	---	------	---	-----	-----	-----------	--	--

2.7.1 Report 1: rptEmployeeCard

Purpose: To display Employee Information to gain access to Staff only areas and to prove you are employed by BelTel, also acts as a name badge for customers.

Data Source: tblEmployee

Formatting:

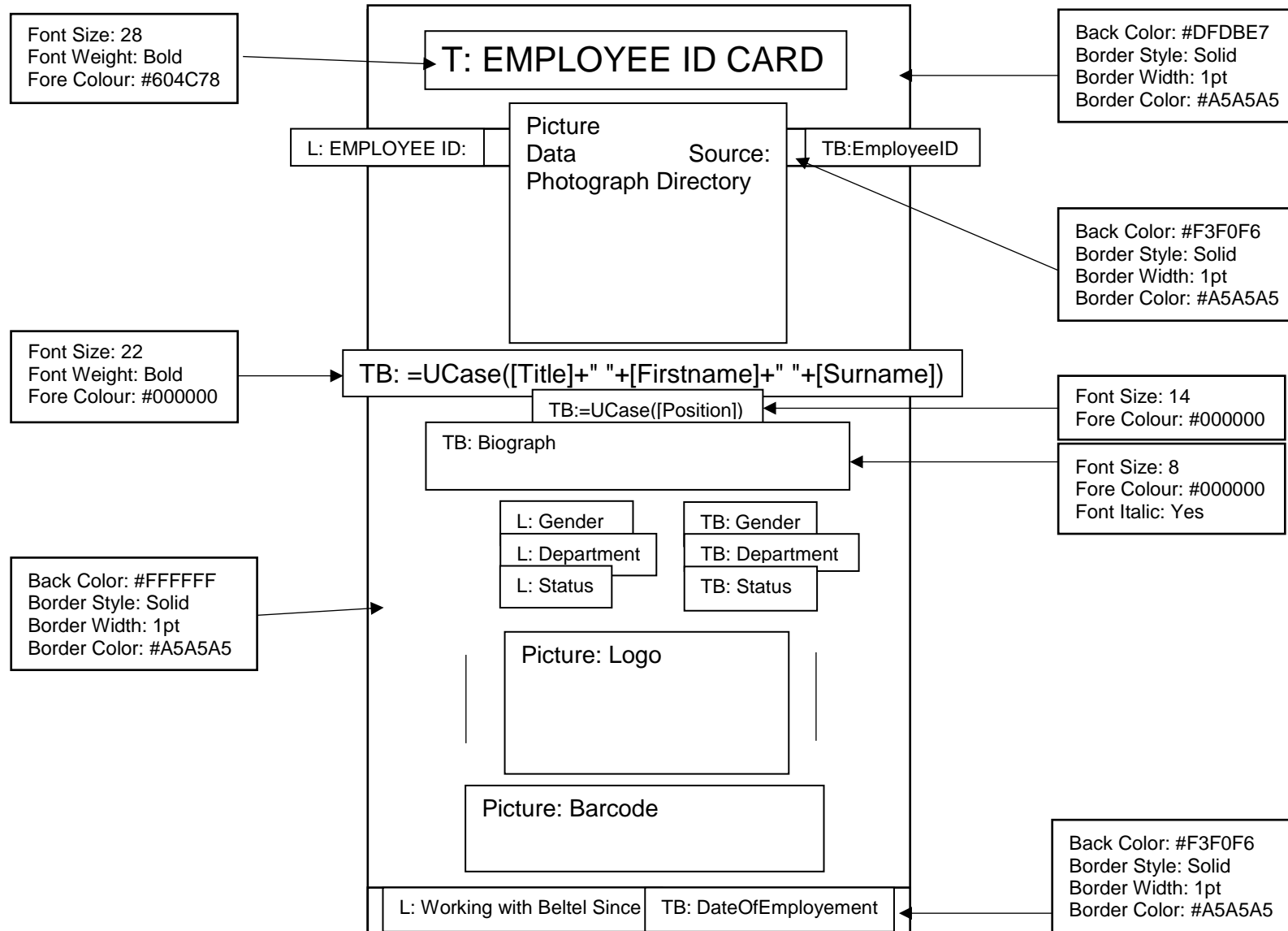
Labels

Font Name:	<i>Calibri (Detail)</i>
Font Size:	<i>11</i>
Font Weight:	<i>Normal</i>
Fore Color:	<i>Text 1, Lighter 50%</i>
Border Style:	<i>Transparent</i>
Text Align:	<i>Left</i>

Text Box

Border Style:	<i>Transparent</i>
Font Name:	<i>Calibri (Detail)</i>
Font Size:	<i>11</i>
Font Weight:	<i>Normal</i>
Fore Color:	<i>#000000</i>
Text Align:	<i>General</i>

Label:



2.7.2 Report 2: rptMembershipCard

Purpose: To gain access to BelTels facilities, members will need to carry their Membership Card as proof, also displays general information for staff.

Data Source: tblMembers, tblMemberMembership, tblMembership

Formatting:

Labels

Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *Text 1, Lighter 50%*
Border Color: *No Color*
Text Align: *Left*

Text Box

Border Style: *Transparent*
Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *#000000*
Text Align: *General*

Label:

T: MEMBERSHIP CARD			
L: MEMBER ID:	TB: MemberID	Picture Data Source: Photograph Directory	
L: NAME:	TB: =UCase([Title]+ " "+[Firstname]+ " "+[Surname])		
L: GENDER:	TB: =UCase([Gender])		
L: DOB:	TB: DOB		
L: MEMBERSHIP TYPE:	TB: =UCase([TypeOfMember])		
L: MEMBERSHIP LEVEL:	TB: =UCase([MembershipLevel])		
L: JOIN DATE:	TB: DateJoined	L: EXPIRATION DATE:	TB: ExpirationDate

Font Size: 28
Font Weight: Bold
Fore Colour: #604C78

Back Color: #FFFFFF
Border Style: Solid
Border Width: 1pt
Border Color: #A5A5A5

Back Color: #DFDBE7
Border Style: Solid
Border Width: 1pt
Border Color: #A5A5A5

Back Color: #F3F0F6
Border Style: Solid
Border Width: 1pt
Border Color: #A5A5A5

Back Color: #F3F0F6
Border Style: Solid
Border Width: 1pt
Border Color: #A5A5A5

2.7.3 Report 3: rptReceipt

Purpose: This is needed as a proof of purchase after any booking, it displays how much they payed and why they are paying it incase they are mischarged or would like a refund.

Data Source: qryBookingLookup

Formatting:

Back Color: #FFFFFF

Labels & Text Boxes

Font Name: Calibri (Detail)

Font Size: 11

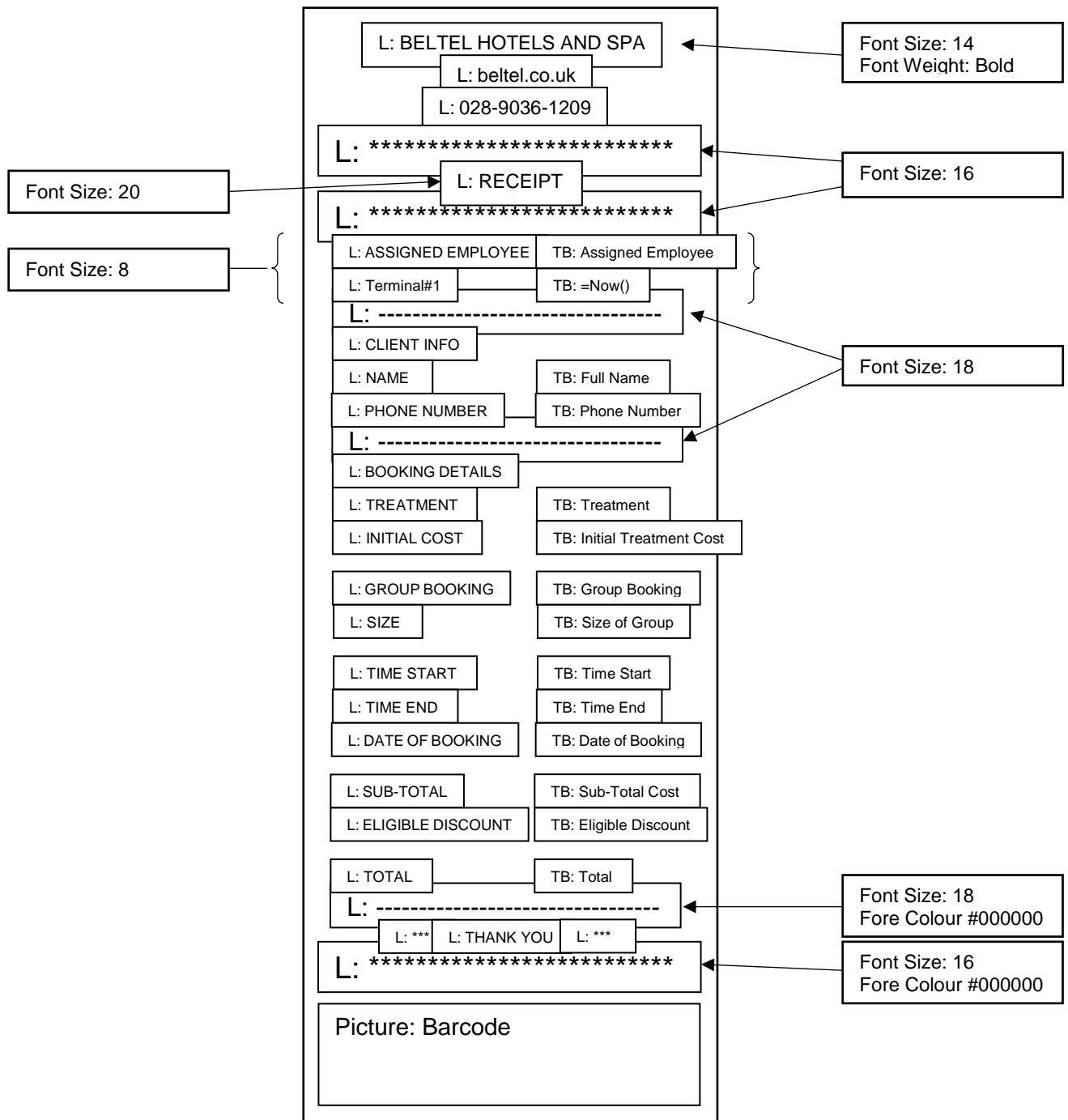
Font Weight: Normal

Fore Color: #000000

Border Color: No Color

Text Align: Left

Label:



2.7.4 Report 4: rptInactiveTemporaryMembers

Purpose: Lists all Temporary Members who have not renewed their membership after the 28 day trail period

Data Source: qryTemporaryMembersNotRenewed

Grouping: TypeOfMember



Sorting: Days Since Joined Descending

Calculations: Total Number of Current Inactive Temporary Mebers

Labels Edited: N/A

Page Breaks: N/A

Report Header:

LG	T: Inactive Temporary Members Report	 
----	--------------------------------------	---

Page Header:

L: Type Of Member	L: Member ID	L: Full Name	L: Phone Number	L: Email Address	L: Membership Level	L: Subscription Plan	L: Days Since Joined	
-------------------	--------------	--------------	-----------------	------------------	---------------------	----------------------	----------------------	--

Group Header:

TB: Type Of Member	
--------------------	--

Detail:

	TB: Member ID	TB: Full Name	TB: Phone Number	TB: Email Address	TB: Membership Level	TB: Subscription Plan	TB: Days Since Joined
--	---------------	---------------	------------------	-------------------	----------------------	-----------------------	-----------------------

Page Footer:

TB: =Now()		TB: ="Page" & [Page] & " of " & [Pages]
------------	--	---

Page Footer:

	L: Total Number of Inactive Temporary Members	TB: =Count(*)
--	---	---------------

2.7.5 Report 5: rptWeeklyBookings

Purpose: Shows Employees all their scheduled bookings within the current week.

Data Source: qryBookingsWeekly

Grouping: Assigned Employee



Sorting: Date Of Booking Ascending

Calculations: Total Number of Bookings per Employee

Labels Edited: N/A

Page Breaks: N/A

Report Header:

LG	T: Weekly Booking Rota	 
----	------------------------	---

Page Header:

L: Assigned Employee	L: Booking ID	L: Full Name	L: Phone Number	L: Treatment	L: Treatment Duration	L: Group Booking	L: Size Of Group	L: Date Of Booking	L: Time Start	L: Time End	
----------------------	---------------	--------------	-----------------	--------------	-----------------------	------------------	------------------	--------------------	---------------	-------------	--

Group Header:

TB: Assigned Employee	L: Total Number of Bookings per Employee:	TB: =Count(*)
-----------------------	---	---------------

Page Header:

	TB: Booking ID	TB: Full Name	TB: Phone Number	TB: Treatment	TB: Treatment Duration	CKB: Group Booking	TB: Size Of Group	TB: Date Of Booking	TB: Time Start	TB: Time End
--	----------------	---------------	------------------	---------------	------------------------	--------------------	-------------------	---------------------	----------------	--------------

Page Footer:

TB: =Now()	TB: ="Page" & [Page] & " of " & [Pages]
------------	---

Page Footer:

L: Total Number of Bookings:	TB: =Count(*)
------------------------------	---------------

2.7.6 Report 6: rptOutstandingPayments

Purpose: Lists all members who have outstanding payments so that BelTel can use their contact information to retrieve what is owed.

Data Source: qryOutstandingPayments

Grouping: Type Of Member



Sorting: Amount Outstanding Descending

Calculations: Total Number of Members per Level who have an Amount Outstanding and Total amount owed

Labels Edited: N/A

Page Breaks: N/A

Report Header:

LG	T: Members with Outstanding Payments	 
----	--------------------------------------	---

Page Header:

L: Type Of Member	L: Member ID	L: Full Name	L: Phone Number	L: Email Address	L: Membership Level	L: Subscription Plan	L: Amount Outstanding	
-------------------	--------------	--------------	-----------------	------------------	---------------------	----------------------	-----------------------	--

Group Header:

TB: Type Of Member		L: Total Number of Members per Level who have an Amount Outstanding:	TB: =Count([Amount Outstanding])
--------------------	--	--	----------------------------------

Detail:

	TB: Member ID	TB: Full Name	TB: Phone Number	TB: Email Address	TB: Membership Level	TB: Subscription Plan	TB: Amount Outstanding
--	---------------	---------------	------------------	-------------------	----------------------	-----------------------	------------------------

Page Footer:

TB: =Now()		TB: ="Page" & [Page] & " of " & [Pages]
------------	--	---

Page Footer:

	L: Total amount owed:	TB: =Sum([Amount Outstanding])
--	-----------------------	--------------------------------

2.7.7 Report 7: rptBookingQuarterlyEarnings

Purpose: View the total amount of earning for any given quarter including the amount of profit from each customer and the total amount of profit.

Data Source: qryBookingQuarterlyEarnings

Grouping: Date Of Booking



Sorting: Full Name Ascending

Calculations: Total amount earned in this Quarter and Total amount earned.

Labels Edited: N/A

Page Breaks: N/A

Report Header:

LG	T: Quarterly Booking Earnings Reports	 
----	---------------------------------------	---

Page Header:

L: Date Of Booking	L: Member ID	L: Full Name	L: Phone Number		L: Total
--------------------	--------------	--------------	-----------------	--	----------

Group Header:

TB: Date Of Booking		L: Total Amount earned in this Quarter:	TB: =Sum([Total])
---------------------	--	---	-------------------

Page Header:

	TB: Member ID	TB: Full Name	TB: Phone Number		L: Total
--	---------------	---------------	------------------	--	----------

Page Footer:

TB: =Now()		TB: ="Page" & [Page] & " of " & [Pages]
------------	--	---

Page Footer:

	L: Total Amount earned:	TB: =Sum([Total])
--	-------------------------	-------------------

2.7.8 Report 8: rptFrequentBookingCancellations

Purpose: Lists members who are frequently cancel bookings incase BelTel needs to penalies them.

Data Source: qryFrequentlyCancelledBookings

Grouping: Membership Level



Sorting: No. Of Cancelled Bookings Descending

Calculations: Total Number of Member per Level who Frequently Cancel and Total Number of Cancelled Bookings

Labels Edited: N/A

Page Breaks: N/A

Report Header:

LG	T: Frequently Cancelled Bookings Report	 
----	---	---

Page Header:

L: Membership Level	L: Type Of Member	L: Member ID	L: Full Name	L: DOB	L: Phone Number	L: Email Address	L: Address	L: Subscription Plan	L: No. Of Cancelled Bookings	
---------------------	-------------------	--------------	--------------	--------	-----------------	------------------	------------	----------------------	------------------------------	--

Group Header:

TB: Membership Level	L: Total Number of Members per Level who Frequently Cancel:	TB: =Count([No. Of Cancelled Bookings])
----------------------	---	---

Page Header:

	TB: Type Of Member	TB: Member ID	TB: Full Name	TB: DOB	TB: Phone Number	TB: Email Address	TB: Address	TB: Subscription Plan	TB: No. Of Cancelled Bookings	
--	--------------------	---------------	---------------	---------	------------------	-------------------	-------------	-----------------------	-------------------------------	--

Page Footer:

TB: =Now()		TB: ="Page" & [Page] & " of " & [Pages]
------------	--	---

Page Footer:

	L: Total Number of Cancelled Bookings:	TB: =Sum([No. Of Cancelled Bookings])
--	--	---------------------------------------

2.7.9 Report 9: rptInvoice

Purpose: The invoice is needed as a record of the sale, it specifies the booking details including costs.

Data Source: tblMembers, qryBookingLookup

Grouping: BookingID

Sorting: N/A

Calculations: N/A

Labels Edited: N/A

Page Breaks: N/A

Report Header:

LG

T: BelTel Booking Invoice

TB: [tblMembers]![Firstname]+" "+[tblMembers]![Surname]

TB: [tblMembers]![Address]

TB: [tblMembers]![Town]

TB: [tblMembers]![County]

TB: [tblMembers]![Postcode]

L: BelTel Hotel & Spa

L: 1 Donegall Lane

L: Belfast

L: Antrim

L: BT1 2LZ

Page Header:

L: BookingID	L: Full Name	L: Phone Number	L: Treatment	L: Initial Treatment Cost	L: Treatment Duration	L: Assigned Employee	L: Group Booking	L: Size Of Group	L: Date Of Booking	L: Time Start	L: Time End	L: Sub-Total	L: Eligible Discount	L: Total	
--------------	--------------	-----------------	--------------	---------------------------	-----------------------	----------------------	------------------	------------------	--------------------	---------------	-------------	--------------	----------------------	----------	--

Group Header:

TB: BookingID	
---------------	--

Page Header:

TB: BookingID	TB: Full Name	TB: Phone	TB: Treatment	TB: Initial Treatment Cost	TB: Treatment Duration	TB: Assigned Employee	CKB: Group Booking	TB: Size Of Group	TB: Date Of Booking	TB: Time Start	TB: Time End	TB: Sub-Total	TB: Eligible Discount	TB: Total	
---------------	---------------	-----------	---------------	----------------------------	------------------------	-----------------------	--------------------	-------------------	---------------------	----------------	--------------	---------------	-----------------------	-----------	--

Page Footer:

TB: =Now()		TB: ="Page" & [Page] & " of " & [Pages]
------------	--	---



Page Footer:

	L: You Saved a Total Of:	TB: Eligible Discount
--	--------------------------	-----------------------

2.7.10 Report 10: rptInactiveMembers

Purpose:
Data Source: qryInactiveMembers
Grouping: Months since Membership Expired
Sorting: Full Name Ascending
Calculations: N/A
Labels Edited: N/A
Page Breaks: N/A

Report Header:

LG	T: Inactive Members Report	 
----	----------------------------	---

Page Header:

L: Months since Membership Expired	L: Member ID	L: Full Name	L: Phone Number	L: Email Address	L: Type Of Member	L: Membership Level	L: Subscription Plan	
------------------------------------	--------------	--------------	-----------------	------------------	-------------------	---------------------	----------------------	--

Group Header:

TB: Months since Membership Expired	
-------------------------------------	--

Page Header:

	TB: Member ID	TB: Full Name	TB: Phone Number	TB: Email Address	TB: Type Of Member	TB: Membership Level	TB: Subscription Plan	
--	---------------	---------------	------------------	-------------------	--------------------	----------------------	-----------------------	--

Page Footer:

TB: =Now()		TB: ="Page" & [Page] & " of " & [Pages]
------------	--	---

Page Footer:

--

2.8 Macro Design

2.8.1 Marco 1: AutoExec

Purpose: This will automatically execute instructions upon the opening of the database, specifically this is needed to minimise the navigation panel so the user cant see the object list incase they accidentally edit anything and to open up a switchboard so they can properly navigate the system with ease.

Frequency: Opening of the Database

MinimizeWindow		
LockNavigationPane	Lock	Yes
OpenForm	Form Name	Switchboard
	View	Form
	Window Mode	Normal

2.8.2 Macro 2: mcrArchiveAllRecords

Purpose: Archives every eligible record throughout the entire database

Frequency: Adhoc

RunMacro	Macro Name	mcrArchiveBookingDetails
	Repeat Count	
	Repeat Expression	
RunMacro	Macro Name	mcrArchiveMemberDetails
	Repeat Count	
	Repeat Expression	
RunMacro	Macro Name	mcrArchiveEmployeeDetails
	Repeat Count	
	Repeat Expression	

2.8.3 Marco 3: mcrArchiveBookingDetails

Purpose: Archive all records relating to a Booking that meets the requirements to be archived.

Frequency: Adhoc

OpenQuery	Query Name	qryAppArchiveBookings
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedBookings
	View	Datasheet
	Data Mode	Edit

2.8.4 Marco 4: mcrArchiveMemberDetails

Purpose: Archive all records relating to a Member that meets the requirements to be archived.

Frequency: Ahoc

OpenQuery	Query Name	qryAppArchiveMemberMedical
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedMemberMedical
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryAppArchivePaymentMethod
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedPaymentMethod
	View	Datasheet
	Data Mode	Edit

OpenQuery	Query Name	qryAppArchiveMedical
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedMedical
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryAppArchivePayments
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedPayments
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryAppArchiveMemberMembership
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedMemberMembership
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryAppArchiveMembers
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedMembers
	View	Datasheet
	Data Mode	Edit

2.8.5 Marco 5: mcrArchiveEmployeeDetails

Purpose: Archive all records relating to an Employee that meets the requirements to be archived.

Frequency: Ahoc

OpenQuery	Query Name	qryAppArchiveEmployee
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryDelArchivedEmployee
	View	Datasheet
	Data Mode	Edit

2.8.6 Marco 6: mcrBackupDatabase

Purpose: Creates a Backup the database incase the original file gets corrupted or destroyed.

Frequency: Adhoc

CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblBookingsB
	Source Object Type	Table
	Source Object Name	tblBookings
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblEmployeeB
	Source Object Type	Table
	Source Object Name	tblEmployee

CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblMedicalB
	Source Object Type	Table
	Source Object Name	tblMedical
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblMemberMedicalB
	Source Object Type	Table
	Source Object Name	tblMemberMedical
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblMemberMembershipB
	Source Object Type	Table
	Source Object Name	tblMemberMembership
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblMembersB
	Source Object Type	Table
	Source Object Name	tblMembers
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblMembershipLevelB
	Source Object Type	Table
	Source Object Name	tblMembershipLevel
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblPaymentMethodB
	Source Object Type	Table
	Source Object Name	tblPaymentMethod
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblPaymentsB
	Source Object Type	Table
	Source Object Name	tblPayments
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblSubscriptionPlanB
	Source Object Type	Table
	Source Object Name	tblSubscriptionPlan

CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	tblTreatmentsB
	Source Object Type	Table
	Source Object Name	tblTreatments
Continue in similar format for Archived Tables		
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	qryAppArchiveBookingsB
	Source Object Type	Query
	Source Object Name	qryAppArchiveBookings
Continue in similar format for the rest of the Queries		
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	frmBookingB
	Source Object Type	Form
	Source Object Name	frmBooking
Continue in similar format for the rest of the Forms		
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	rptWeeklyBookingsB
	Source Object Type	Report
	Source Object Name	rptWeeklyBookings
Continue in similar format for the rest of the Reports		
CopyObject	Destination Database	E:\Year 14\Digital Technology\Mrs Nevin\5087 JK Backup.ACCDB
	New Name	mcrArchiveAllRecordsB
	Source Object Type	Macro
	Source Object Name	mcrArchiveAllRecords
Continue in similar format for the rest of the Macros		

2.8.8 Marco 7: mcrUpdateExpiredMembers

Purpose: Updates all members who are eligible to be archived so the system can remain up to date and the information currently on the system is useful.

Frequency: Adhoc

OpenQuery	Query Name	qryUpdTemporaryMembers
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryUpdExpiredMembers
	View	Datasheet
	Data Mode	Edit

2.8.9 Marco 8: mcrUpdateBookingEndTime

Purpose: Updates the time at which the booking will end upon based on the time at which it starts and the treatments duration, it will commence upon an action in the form.

Frequency: Adhoc

OpenQuery	Query Name	qryUpdBookingEndTime
	View	Datasheet
	Data Mode	Edit

2.8.9 Marco 9: mcrUpdateBookingStatus

Purpose: Enables the employee to either update the status of a booking to cancelled or completed, if the status is changed to completed it will add a loyalty point to a given member.

Frequency: Adhoc

OpenQuery	Query Name	qryUpdCompleteBooking
	View	Datasheet
	Data Mode	Edit
OpenQuery	Query Name	qryUpdLoyaltyPoints
	View	Datasheet
	Data Mode	Edit

2.9 Switchboard Design

2.9.1 Switchboard Formatting

Form Header Formatting:

Back Color: #DFDBE7

Title

Font Name: Calibri (Detail)

Font Size: 36

Font Weight: Bold

Fore Color: #604C78

Border Color: No Color

Text Align: General

Title (Label 2)

Visible: No

Logo

Name: Beltel Logo Cropped

Location: D:\Year 14\Digital Technology\Mrs Nevin\Portfolio Data

File Type: PNG

Width: 4.157cm

Height: 2.083cm

Body Formatting:

Back Color: #FFFFFF

Alternate Back Color: #F2F2F2

Box

Back Color: #F3F0F6

Border Style: Transparent

Line

Border Style: Solid

Border Width: 3 pt

Border Color: #6F3198

Represented by: _____

Command Button

Back Color: #6F3198

Hover Color: #9F8CB7

Pressed Color: #604C78

Option Labels

Font Name: *Calibri (Detail)*
Font Size: *11*
Font Weight: *Normal*
Fore Color: *#9F8CB7*
Border Color: *No Color*
Text Align: *General*
Width: *7.037cm*
Height: *0.476cm*

Key

T	Title	LG	Logo	L	Label	B	Box	OL	Option Label	CDB	Command Button
---	-------	----	------	---	-------	---	-----	----	--------------	-----	----------------

Form Header:

LG	T: Main Switchboard	T (L2)
----	---------------------	--------

Detail:

B	CDB	OL: ItemText
---	-----	--------------

2.9.2 Switchboard Content

Main Switchboard

1. *Members Menu*
2. *Employees Menu*
3. *Bookings Menu*
4. *Treatments Menu*
5. *Archive All Records*
 - Runs the macro *mcrArchiveAllRecords*
6. *Run Backup*
 - Runs the macro *mcrBackupDatabase*
7. *Exit Application*

Members Menu

1. *Add New Member*
 - Opens the form *frmMembers* in *Add Mode*
2. *Edit Member Details*
 - Opens the form *frmMembers* in *Edit Mode*
3. *Update Members Eligible for Archive*
 - Runs the macro *mcrUpdateExpiredMembers*
4. *Membership Level Menu*
5. *Member Lookup Menu*
6. *Member Reports Menu*
7. *Archive Member Records*
 - Runs the macro *mcrArchiveMemberDetails*
8. *Main Menu*

Membership Level Menu

1. *Add New Membership Level*
 - Opens the form *frmMembership* in *Add Mode*
2. *Edit Membership Level Details*
 - Opens the form *frmMembership* in *Edit Mode*
3. *Member Menu*

Member Lookup Menu

1. *Lookup Members by Name*
 - Runs the query *qryMembersLookup*
2. *Lookup Members by Address*
 - Runs the query *qryMemberAddressLookup*
3. *Lookup Members by Payment Details*
 - Runs the query *qryMemberPaymentLookup*
4. *Lookup Members by Medical Records*
 - Runs the query *qryMemberMedicalRecords*
5. *List of Members and the Total Amount They've Spent on Bookings*
 - Runs the query *qryMembersBookingSpent*
6. *Member Menu*

Member Reports Menu

1. *Membership Card*
 - Opens the report *rptMembershipCard*
2. *Inactive Temporary Members Report*
 - Opens the report *rptInactiveTemporaryMembers*
3. *Members with Outstanding Payments Report*
 - Opens the report *rptOutstandingPayments*
4. *Frequently Cancelled Bookings Report*
 - Opens the report *rptFrequentBookingCancellations*
5. *Member Menu*

Employees Menu

1. *Add New Employee*
 - Opens the form *frmEmployee* in *Add Mode*
2. *Edit Employee Details*
 - Opens the form *frmEmployee* in *Edit Mode*
3. *Employee Lookup Menu*
4. *Employee Reports Menu*
5. *Archive Employee Records*
 - Runs the macro *mcrArchiveEmployeeDetails*
6. *Main Menu*

Employee Lookup Menu

1. *Lookup Employees by Name*
 - Runs the query *qryEmployeeLookup*
2. *Lookup Employees by Address*
 - Runs the query *qryEmployeeAddressLookup*
3. *List an Employees Total Amount Earned from Bookings*
 - Runs the query *qryEmployeeBookingEarnings*
4. *List of Bookings Designated to an Employee*
 - Runs the query *qryEmployeeBookings*
5. *Employee Menu*

Employee Reports Menu

1. *Employee ID Cards*
 - Opens the report *rptEmployeeCard*
2. *Employee Menu*

Bookings Menu

1. *Add New Booking*
 - Opens the form *frmBooking* in *Add Mode*
2. *Edit Booking Details*
 - Opens the form *frmBooking* in *Edit Mode*
3. *Update Booking Status*
 - Runs the macro *mcrUpdateBookingStatus*

4. *Booking Lookup Menu*
5. *Booking Reports Menu*
6. *Archive Booking Records*
 - Runs the macro *mcrArchiveBookingDetails*
7. *Main Menu*

Bookings Lookup Menu

1. *Lookup Bookings by ID*
 - Runs the query *qryBookingLookup*
2. *List of Bookings in the Current Week*
 - Runs the query *qryBookingsWeekly*
3. *Bookings Menu*

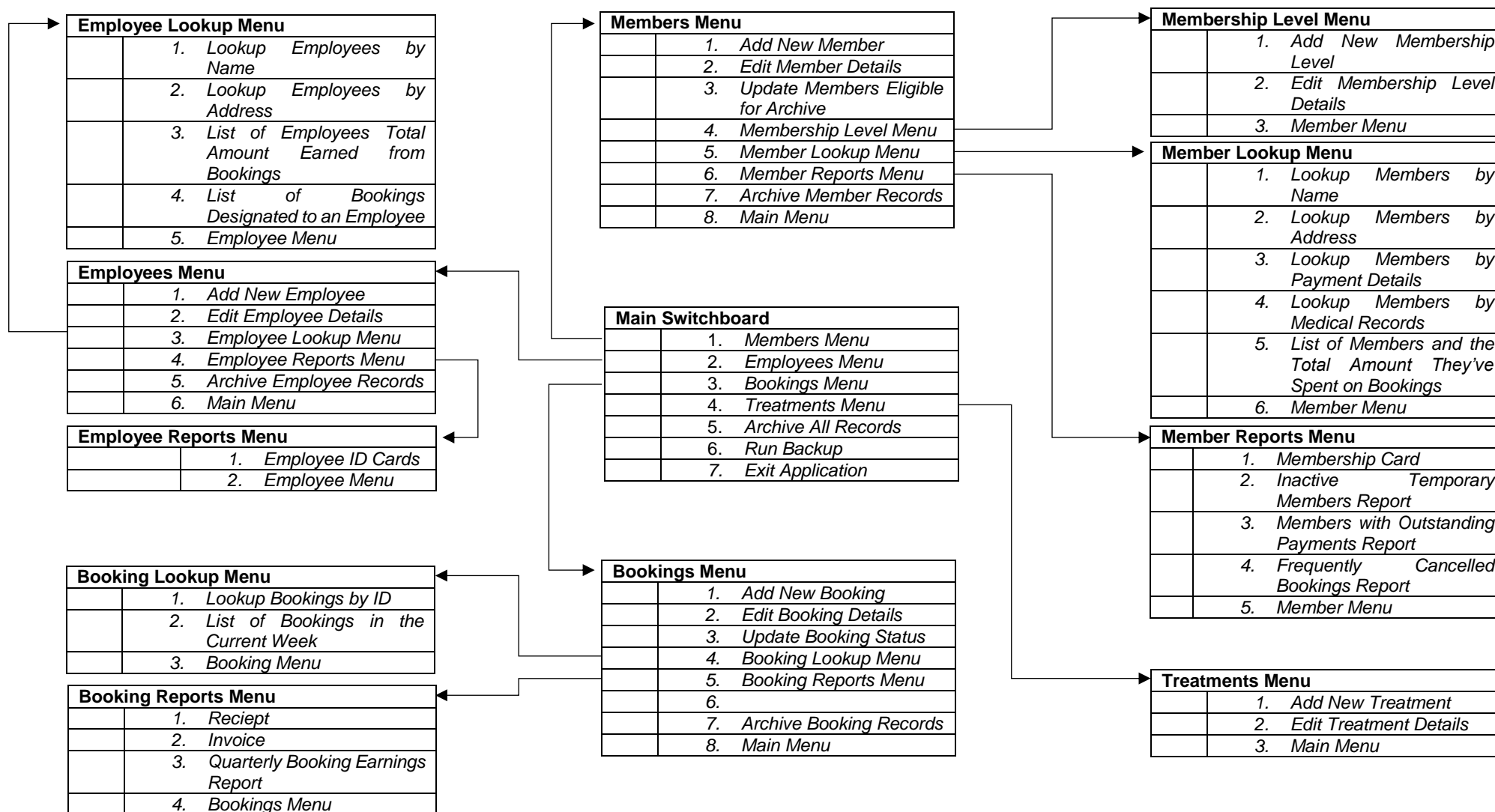
Bookings Reports Menu

1. *Reciept*
 - Opens the report *frmReciept*
3. *Invoice*
 - Opens the report *frmInvoice*
4. *Quarterly Booking Earnings Report*
 - Opens the report *frmBookingQuarterlyEarnings*
5. *Bookings Menu*

Treaments Menu

1. *Add New Treatment*
 - Opens the form *frmTreatment* in *Add Mode*
2. *Edit Treatment Details*
 - Opens the form *frmTreatment* in *Edit Mode*
3. *Main Menu*

2.9.3 Switchboard Navigation



Chapter 3 – Application Development

3.1 Unit Testing

Unit Testing is carried out under teacher observation.

3.2 User Requirement Testing

Test No	Req No	Area of System	Test Type	Method of Testing	Test Data Used	Expected Outcome	Actual Outcome & Corrective Action

3.3 Corrective Action

3.2.1 CA#1

3.2.2 CA#2

- Problem with combo boxes in frmBooking with drop down list for Member Name, Treatments and Employee Name. (Not Solved)

3.2.3 CA#3

- 3.2.4 CA#4 Initial problems with Option Groups in frmEmployee, used code to solve it.

```
Private Sub frmTitle_AfterUpdate()  
Select Case frmTitle.Value  
Case 1  
Title = Null  
Case 2  
Title = "Mr"  
Case 3  
Title = "Mrs"  
Case 4  
Title = "Miss"  
Case 5  
Title = "Dr"  
Case 6  
Title = "Rev"  
End Select  
End Sub
```

```

Private Sub Form_Current()
Select Case Title.Value
Case ""
frmTitle.Value = 1
Case "Mr"
frmTitle.Value = 2
Case "Mrs"
frmTitle.Value = 3
Case "Miss"
frmTitle.Value = 4
Case "Dr"
frmTitle.Value = 5
Case "Rev"
frmTitle.Value = 6
End Select

```

3.2.5 CA#5

- Problem with updating End Time, tried to use a select query calculation, instead used an update query and changed values in the Duration field in tblTreatments

Field:	EndTime
Table:	tblBookings
Update To:	[tblBookings]![TimeSt
Criteria:	
or:	

3.2.6 CA#6

- Problem updating member loyalties, initially tried to use a select query to count the no of bookings per member and update it to the Member table.

Field:	Loyalty
Table:	tblMembers
Update To:	[qryNoBookingsPerMember].[CountOfBookingID]
Criteria:	
or:	

Field:	Loyalty	MemberID
Table:	tblMembers	tblMembers
Update To:	[Loyalty] + 1	
Criteria:		[frmBooking]![MemberID]

3.2.7 CA#7

- Initially had payment details entered in the Membership form as well as medical information, this didn't let me edit member details who didn't already have a medical record. Also couldn't have multiple records per member.

<h1>Payment Details Form</h1>															
				Payment Method ID: <input type="text" value="PaymentN"/>					Search: <input type="text" value="Unbound"/>						
Detail															
<h2>Payment Details</h2>															
Payment Method		<input type="text" value="PaymentMethc"/>		Payment Type		<input type="text" value="PaymentType"/>		Status		<input type="text" value="Status"/>					
Card Number		<input type="text" value="CardNumber"/>		Expiration Year		<input type="text" value="Expir"/>		Expiration Month		<input type="text" value="Expir"/>		Security Number		<input type="text" value="Securit"/>	
Form Footer															
<input type="button" value="Add New Payment Record"/>								<input type="button" value="Delete Payment Record"/>							

<h1>Medical Form</h1>															
				Medical ID: <input type="text" value="MedicalID"/>					Search: <input type="text" value="Unbound"/>						
Detail															
<h2>Medical Details</h2>															
Condition Name		<input type="text" value="ConditionName"/>													
Restrictions		<input type="text" value="Restrictions"/>													
Form Footer															
<input type="button" value="Add New Medical Record"/>								<input type="button" value="Delete Medical Record"/>							

3.4 Acceptance Testing

3.5 Strategy for System Implementation

This topic is needed in order cover how the old system will be replaced by the new system and as well as how it will be introduced. Various things will need to be discussed such as the different resources required like hardware and software which will need to be in place prior to the new system being introduced. Furthermore, converting data and the method of changeover are necessary to be decided upon and completed before the final system is in-place and operational, questions may be of great use to decide upon the method of changeover.

Once the technical aspect of system implementation is completed, training will need to go underway. Two suitable methods of training must be decided upon prior to this commencing; a backup plan for COVID restrictions would also be ideal.

3.5.1 Hardware and Software Installation

Hardware currently in use at BelTel is not sufficient enough to meet the new requirements established in order to operate the system in an effective and efficient manner. Therefore, new hardware must be purchased, and old hardware updated to meet the minimum specification underlined in “[1.5.1 Hardware](#)” which will allow the system to work at optimum capacity.

BelTel will require a minimum of **four** new workstations to be installed on site; one for the reception area to manage bookings and member details, another for the treatment facility to amend bookings and a third workstation for HR to manage employees, treatment and membership details. **The final workstation will be utilized as a file server for the other workstations.** Each workstation will require a single QWERTY keyboard and mouse. A minimum of two printers would also be required for both HR and the reception desk in order to print of things such as employee and customer forms as well as invoices. Finally, a single payment terminal would be required at the reception desk to print receipts, process payments and deposit money into.

To effectively carry out BelTels’ operations, a new file server will be required in order to establish a Local Area Network, which in return will enable a company intranet to be organized. The file server is the main computer on the network with a large amount of RAM and hard disk space dedicated to storing user files, it will store other things such as the application, system and utility software which is far more effective than installing it individually on each workstation.

Furthermore, a switched hub would be another component required to allow multiple users to use the system simultaneously without data collisions. It acts as a connection point for several network computers or nodes, it checks the IP address of data packets and forwards them to the intended recipient.

(Are these in HW def covered In 1.5.1? if so reword)

Following this, **the site will need to be properly cabled as the system currently in use is paper-based, therefore there is no prior cabling to utilize.** An appropriate layout for the cabling and nodes must be chosen which is referred to as the network topology. The most effective network topology for this scenario would be a star topology; Each network device or node will have a separate connection to the file server. It has been selected as the typology of choice as they are very reliable; if one connection were to fail it would not affect others, its secure and simple to add more devices. However, a disadvantage of this layout is that if the fileservers were to malfunction, then the whole network would fail. This would require the site to have new copper cablings installed in a fashion where it is out of site and secure to keep the facility looking professional and organized.

The hardware will arrive on the 8th of February 2021 and will be given the next 4 days to install it, the deadline for it to have been installed will be the 12th of February 2021

To ensure that the systems performs well during its operations the **software** will need to be updated so it is compatible with the new systems hardware.

The operating system specified is Windows 10 Enterprise Edition which should already come pre-loaded on each workstation, if BelTel has purchased the Dell OptiPlex 7020 as suggested previously. It is a system software program that manages all the operations and resources of a computer. It allows human-computer interaction acting as a bridge between the hardware, application software and its end users through its' graphical user interface that allows input through visual cues, it consists of; Windows, Icons, Menus and Pointers. The specific edition also has additional features such as built-in security which show its advantage over other versions.

Expanding upon this, application software will also be required. Microsoft Access 2016 will need to be installed in order to operate the system being developed for BelTel, it helps with the analysis of large amounts of data, its validation, processing and outputs. It also comes with many other additional features such as visual tools for designing queries, printing off forms, etc. Microsoft Access 2016 has been suggested as the email client in order to establish formal communications between colleagues, Microsoft Excel 2016 to develop spreadsheets to help BelTel compile things such as their financial data in a visual format and Avast Anti-Virus in order to prevent, detect and remove malware, this will add an extra layer of security protection on top of Windows built in security systems. All application software will be installed solely onto the file **server** to save much needed time. The system being developed for BelTel will also need to be installed alongside the application software.

The software **apart from the bespoke application for BelTel** will arrive on the 12th of February 2021 and will be given the next 3 days to install it, the deadline for it to have been installed will be the 15th of February 2021.

3.5.2 Data Conversion

Data conversion is the process of converting or transferring data from the old system to the new one. This process is needed as existing data may not be structured in a way that matches what will be found in the new system and will be required in order to populate the new system with data.

The current system utilizes paper-based records consisting of forms, reports, invoices and files all of which will have to be reorganised to suit the new databases structure. Following this the non-electronic data will have converted to electronic data and keyed in, **validated by the software and then** verified by human. Great care must be taken during this process to ensure all necessary data is converted correctly in the process.

(Is your application developed at this point to allow for it?)

Data Conversion will commence on the **22nd of February**, each day a part of the system will be fully converted in order to phase parts of the system in. It will be given a total of 6 days to be completed giving it a deadline of the 28th of February, this is done in order to complete the migration of the entire database from one format to another.

3.5.3 Changeover Method

Changeover is the process of starting to use a computer system in the "live" environment after having designed and developed it. It will allow final testing in a real situation if necessary. There are various different approaches that can be used depending on the size of the system and the properties of the data being processed, these methods include; Direct, Phased, Parallel and Pilot changeover.

Changeover describes how the computer system will be used and implemented initially in its "live" environment which is at BelTel after it is completed the various stages of design and development.

Immediately it can be determined that Pilot Changeover is not a suitable method of changeover in this scenario as the new system is only being developed for a business that has one location. Pilot Changeover would require at least one more branch to test the new system and compare it to that of the old system.

To ensure that the most appropriate method of changeover was selected various questions were asked to the end user in order to determine a solution:

Could you afford to close the business for a few days?

“No, BelTel relies financially on its constant operation and closing for a few days would mean a significant loss of capital.” Since the business must remain open during changeover, Direct Implementation is no longer a suitable method as it will take some time to complete data conversion of the current system which utilizes paper-based records in a non-electronic format to the new system using an electronic format.

Could you afford to use both a paper-based and electronic system simultaneously in full operation?

“No, it would take too many resources to duplicate information for both our paper-based records and our digitised records.” Therefore, Parallel Implementation cannot be used as a changeover method as it would be too costly to operate both systems at the same time and double employee workload and reduce their productivity.

Could you afford to additional employee temporary staff?

“Yes, I could employee additional staff temporarily if required to get the new system implemented.” This means extra staff can be employed to help speed up the process of data conversion.

Could you start recording all your member details in the new systems while still having a manual appointment book, this would mean data would not be duplicated as you'd only be using parts of the new system.

“Yes, it shouldn't be hard for our staff to use the new system to enter data on our members whilst still using an appointment book.” This makes Phased Implementation a suitable candidate as a changeover method.

Through the use of carefully crafted questions, the method of changeover has been narrowed down to Phased implementation, that being the most suitable method for BelTel. It will allow time for data conversion to commence alongside the gradual installation of the system, extra staff will also help speed up this process, since it is being installed in stages errors can be detected early on before the system is fully operational and independent. This also means staff training can commence in intervals when certain parts of the system become functional on the new system.

Phase implementation will start from the 23rd of February 2021 and will be allocated 6 days to fully implement the system, the deadline for its completion is the 1st of March 2021.

3.5.4 Training

Staff at BelTel will need to under-go training on how to operate the new systems correctly and safely in order to comply with HASAWA (Health and Safety at Work Act 1974). There are various different methods which can be chosen to train employees, the most appropriate being instructor-led training as well as a user guide. Due to current COVID restrictions, we will also need to cover a backup plan in case instructor-led training cannot be performed and instead will be replaced by videoconferencing if required.

Instructor-led training will take place on each day throughout phased implementation from the 23rd of February 2021 to the 1st of March 2021, lasting a total of 6 days (2 days max) in order to cover each part of the system as it comes into operation, there will be a specific structure to it with there being both morning and afternoon sessions to ensure every employee has the chance to receive training. Instructor led training has been selected as the training method of choice as it allows a high level of interaction, employees will be able to discuss and ask any questions to the instructor to further understand the system and it will also be easier to focus with someone guiding you through it. This session will include a step-by-step walkthrough of each operation that can be performed by the system in question displayed on a projector screen as well as designated time following this to ask questions so individuals can receive help if

needed to allow for further explanation on any issues experienced during their training. A series of practice exercises will be **used** in the course to ensure the employees understand what they are doing and to assess their learning. This will be a cost-effective method of training as staff can be trained in large groups therefore minimising the cost to employee more instructors.

Furthermore, if COVID restrictions prevent instructor-led training from taking place, videoconferencing will be used as a substitute. Similarly to instructor-led training, it will include a step-by-step walkthrough of each operation that can be performed by the system in question as well as designated time following this to ask questions so individuals can receive further explanation on any issues experienced during their training, however, hands-on individual help will not be possible with this method since the instructor won't be physically on site nor will the instructor be able to **assist** the employees **with the practise exercises.**

A detailed user-guide will be provided along with an index to find a specific section. It will include an in-depth step-by-step walkthrough of each operation the database can perform with screenshots being coincided with the text. Furthermore, there will also be a troubleshooting section providing clear explanation on how to deal with any errors you may encounter. Finally, FAQs will also be provided within the final pages of the guide to provide further explanation. It will be included within the database so its easily accessible at any time for future reference and a physical copy will also be distributed to each employee during instructor-led training, if this cannot take place then a PDF file will be supplied to each employee during videoconferencing. This will allow the employees to work through it in their own time and at their own pace to familiarize themselves with its contents. The cost of developing this guide will be a one-time purchase and **is included in the budget.**

3.5.5 Summary

(Dates of when each stage taking place.)

Chapter 4 – Evaluation

4.1 Evaluation of User Requirements

4.1.1 Essential Requirements

E1 Add, Edit and Delete Member Records

This requirement has been met.

2.3.4 Table 4:	tblMemberMembership
2.3.5 Table 5:	tblMembers
2.5.4 Form 4:	frmMember
2.6.5 Query 5:	qryAppArchiveMemberMembership
2.6.6 Query 6:	qryAppArchiveMembers
2.6.13 Query 13:	qryDelArchivedMemberMembership
2.6.14 Query 14:	qryDelArchivedMembers
2.6.31 Query 31:	qryUpdExpiredMembers
2.6.32 Query 32:	qryUpdMemberLoyalties
2.6.33 Query 33:	qryUpdTemporaryMembers

E2 Add, Edit and Delete Payment Records

This requirement has been met.

2.3.7 Table 7:	tblPaymentMethod
2.3.8 Table 8:	tblPayments
2.5.5 Form 5:	frmPayment
2.6.7 Query 7:	qryAppArchivePaymentMethod
2.6.8 Query 8:	qryAppArchivePayments
2.6.15 Query 15:	qryDelArchivedPaymentMethod
2.6.16 Query 16:	qryDelArchivedPayments

E3 Add, Edit and Delete Medical Records

This requirement has been met.

2.3.3 Table 3:	tblMedical
2.3.11 Table 11:	tblMemberMedical
2.5.3 Form 3:	frmMedical
2.6.3 Query 3:	qryAppArchiveMedical
2.6.4 Query 4:	qryAppArchiveMemberMedical
2.6.11 Query 11:	qryDelArchivedMedical
2.6.12 Query 12:	qryDelArchivedMemberMedical

E4 Add, Edit and Delete Employee Records

This requirement has been met.

2.3.2 Table 2:	tblEmployee
2.5.2 Form 2:	frmEmployee
2.6.2 Query 2:	qryAppArchiveEmployee
2.6.10 Query 10:	qryDelArchivedEmployee

E5 Add, Edit and Delete Booking Records

This requirement has been met.

2.3.1 Table 1:	tblBookings
2.5.1 Form 1:	frmBooking
2.6.1 Query 1:	qryAppArchiveBookings
2.6.9 Query 9:	qryDelArchivedBookings
2.6.29 Query 29:	qryUpdBookingEndTime
2.6.30 Query 30:	qryUpdCompleteBooking

E6 Backup Systems

This requirement has been met.

E7 Archive Records

This requirement has been met.

- 2.6.1 Query 1: qryAppArchiveBookings
- 2.6.2 Query 2: qryAppArchiveEmployee
- 2.6.3 Query 3: qryAppArchiveMedical
- 2.6.4 Query 4: qryAppArchiveMemberMedical
- 2.6.5 Query 5: qryAppArchiveMemberMembership
- 2.6.6 Query 6: qryAppArchiveMembers
- 2.6.7 Query 7: qryAppArchivePaymentMethod
- 2.6.8 Query 8: qryAppArchivePayments

E8 Invoice

This requirement has been met.

E9 Receipt

This requirement has been met.

E10 Update Member status after period of 28 days

This requirement has been met.

- 2.6.33 Query 33: qryUpdTemporaryMembers

E11 Members must be 18 or older

This requirement has been met.

- 2.3.5 Table 5: tblMembers

E12 Must be a member to book a treatment

This requirement has been met.

- 2.3.1 Table 1: tblBookings

E13 Group bookings have a minimum size of 5 and maximum size of 10

This requirement has been met.

- 2.3.1 Table 1: tblBookings

E14 Update loyalty point for each booking

This requirement has been met.

- 2.6.32 Query 32: qryUpdMemberLoyalties

E15 Password protected with employee logins

This requirement has been met.

E16 Apply Membership fees depending on membership level

This requirement has been met.

E17 Update discounts on bookings

This requirement has been met.

E18 Employee weekly booking rota

This requirement has been met.

- 2.6.22 Query 22: qryEmployeeBookings

E19 Search for given members

This requirement has been met.

- 2.6.23 Query 23: qryMemberAddressLookup
- 2.6.27 Query 27: qryMemberLookup

E20 Search for given employees

This requirement has been met.

- 2.6.22 Query 22: qryEmployeeLookup
- 2.6.20 Query 20: qryEmployeeAddressLookup

E21 Calculate Employee Treatment Commission

This requirement has been met.

E22 List of people with outstanding payments

This requirement has been met.

E23 Membership Card

This requirement has been met.

E24 Temporary Members who have not renewed their membership report

This requirement has been met.

E25 Quarterly Bookings Report

This requirement has been met.

E26 Members who frequently cancel bookings report

This requirement has been met.

E27 Calculate weekly treatment income

This requirement has been met.

4.1.2 Non-Essential Requirements**N1 Membership Renewal Notification**

This requirement has been met.

N2 Employee Card

This requirement has been met.

N3 Reservation Confirmation

This requirement has been met.

N4 Inactive Members Report

This requirement has been met.

N5 Booking Reminder

This requirement has been met.

N6 Popular Treatments Report

This requirement has been met.

N7 Quarterly Treatment Profit Graph

This requirement has been met.

N8 Yearly Finance Graph

This requirement has been met.

N9 Retrieve top earning employees

This requirement has been met.

4.2 System Limitations**4.3 Possible Extensions to the System**

- Spouse Table to store spouse details when a member selects “Couple” membership type.

- Employee Overtime table to store information on the week and the overtime hours worked.
- Appointment Selection Spreadsheet
- Products