5/12/2017

Final Year Project 2017

Your Office Planner

A dissertation presented for the degree of BSc. (Hons) Software Engineering.

Student ID

School of Computing, Engineering and Mathematics

University of Brighton

Table of Contents

Project Scope

1.1 Aims and Objectives

The ultimate objective of "Your Office Planner" project is to provide a Room Booking software for World of Books employees. The main aim for this project is to allow employees to create meetings at chosen time and selected room. Employee will be able to store the notes in the program and retrieve them.

1.2 Stakeholders

The Stakeholders involved in this project will be World of Books, myself and supervisor and the second reader. World of Books will be the main stake holder as they will be the main users using this software and testing. Stake holders will be giving the requirements for this project.

1.3 Methods of Communication

Communication with the main stakeholder was twice a week at the World of Books company. Where I've been showing the progress on the project to IT Manager and IT Infrastructure manager. Who gave their opinion about the software and changes that should be made to the project.

1.4 Research

I have made a research on three very important parts of my project which are Programming languages, databases and the encryption methods. The programming language maybe is not as important as the other two however still had to check which programming language to use for this project.

1.4.1 Which language to use?

When it comes to application development there are many programming languages which are relevant to writing such application. Each programming language will fill different role and will have advantages, disadvantages and is different in its own right.

Java

Java is one of the best programming languages created. Java is an Object Orientated Language which has a lot of Open Source libraries, great collection of powerful IDEs and has a rich API. Java offers swing for GUI creation which would allow me to create everything for the project. This is why Java is great for GUI creation which my software will use a lot. Following the research, I have found out that there is not a lot of desktop java jobs, there are many more server side jobs however java can be enough to use for a desktop java application if it can meet all of the requirements. Java doesn't have a framework for Java desktop application which is definitely a disadvantage however it can still produce a good desktop application. Definitely Java is the easiest language to write desktop apps and it is great if you want the software to be compatible with all the platforms.

C++

I have chosen to research C++ as my second language as I have previous experience in using this language. C++ is a multipurpose language. C++ has a disadvantage which it's not good if you want your desktop application to be compatible with all of the platforms. This is why the C++ is not good for this project as "Your Office Planner" will be working on Windows, Mac and Linux.

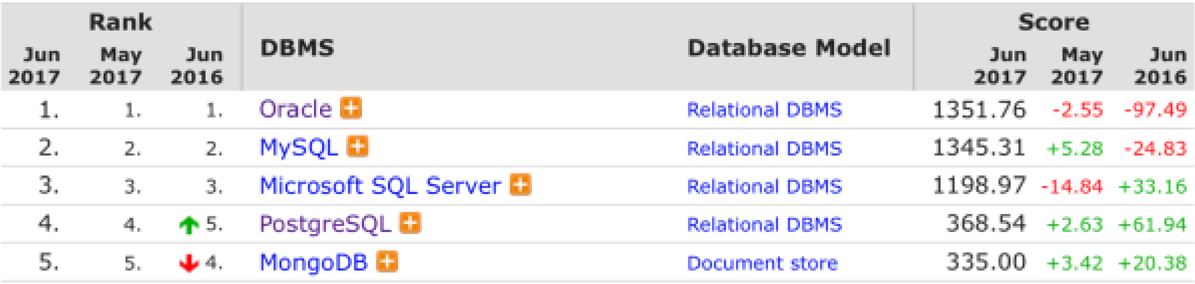
1.4.2 Which Database to use?

SQL vs Postgres

When it comes to applications development where there is sensitive data and a lot of database usage it is very important to choose a right database. The database will hold mostly very confidential information which should not be lost or hacked. The database will continuously create, update and remove data, it will also have encrypted information which will be decrypted and encrypted. MySQL and Postgres are very similar DBMS however there are features and functions that are different and both have their pros and cons.

PostgreSQL has a lot of very useful feature as generate series, custom aggregate functions, arrays and many more which make life a lot easier. However, MySQL has a lot of features that Postgres doesn't have. The main difference but very significant is that MySQL has only one Server operating system which is windows where Postgres has Windows, OS X, Linux and many more it has much broader reach because of that. Postgres is an open source where MySQL is a commercial software. The disadvantage of Postgres definitely has to be that is behind MySQL on the popularity level , what comes with that(See figure 1.4.2) ? It makes it a lot harder to get community support or google the results that you are looking for .

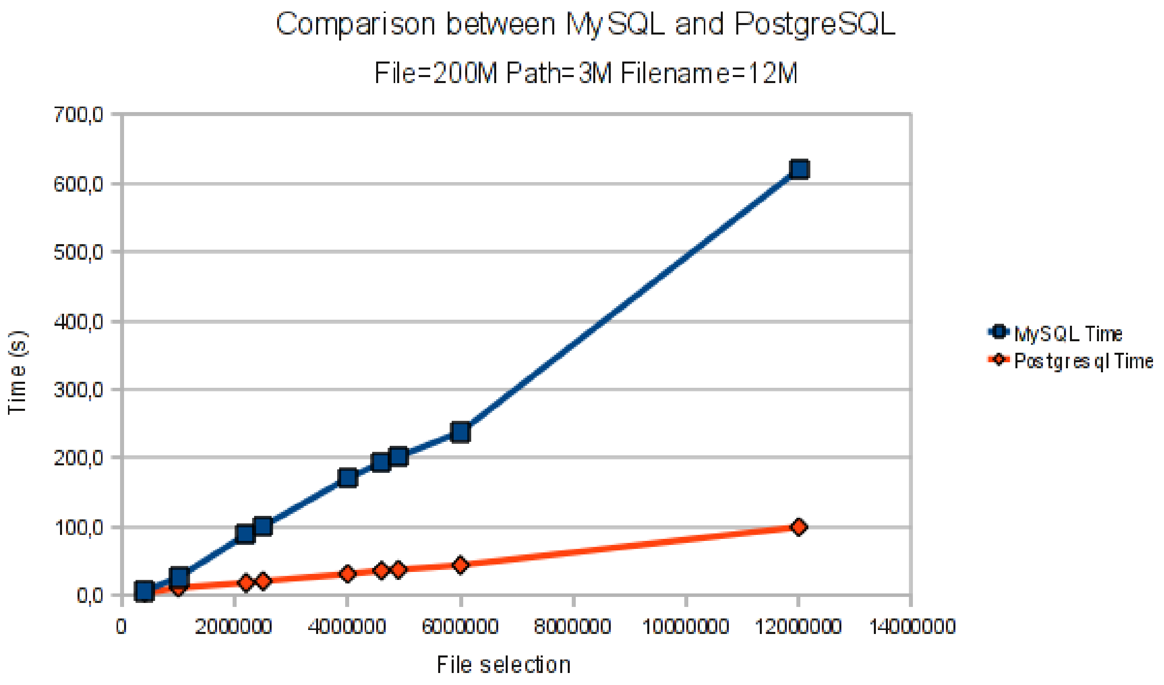
Figure 1.4.2:



Rank			DBMS	Database Model	Score		
Jun 2017	May 2017	Jun 2016			Jun 2017	May 2017	Jun 2016
1.	1.	1.	Oracle ⊞	Relational DBMS	1351.76	-2.55	-97.49
2.	2.	2.	MySQL ⊞	Relational DBMS	1345.31	+5.28	-24.83
3.	3.	3.	Microsoft SQL Server ⊞	Relational DBMS	1198.97	-14.84	+33.16
4.	4.	↑ 5.	PostgreSQL ⊞	Relational DBMS	368.54	+2.63	+61.94
5.	5.	↓ 4.	MongoDB ⊞	Document store	335.00	+3.42	+20.38

Both of the databases have a very good tooling as (MySQL Workbench vs PostgreSQL pgAdmin 3) , have been proven to be mature in production and have companies behind them if you need a professional support(Oracle and Percona vs EnterpriseDB). In terms of performance between the two PostgreSQL is used mostly in large systems where read and write speed is very important and the data has to be validated. MySQL is mostly chosen for the web based applications that need a database for easy and simple data transactions. MySQL usually underperforms when there is heavy loads or running very complex queries. There is a comparison between MySQL and PostgreSQL which show the time difference in file selection between these two DBMS. (See figure 1.4.3)

Figure 1.4.3:



The graph clearly shows the difference between PostgreSQL and MySQL and that the first one is much faster at selecting files. However this would have to be a very big database to see the real time difference between the two. In a smaller queries it doesn't real make this much of a difference.

PostgreSQL has ROLES and inherited roles to set and maintain permissions. PostgreSQL has native SSL support for connections to encrypt client/server communications. It also has Row Level Security MySQL implements security based on Access Control Lists (ACLs) for all connections, queries, and other operations that a user may attempt to perform. There is also some support for SSL-encrypted connections between MySQL clients and servers.

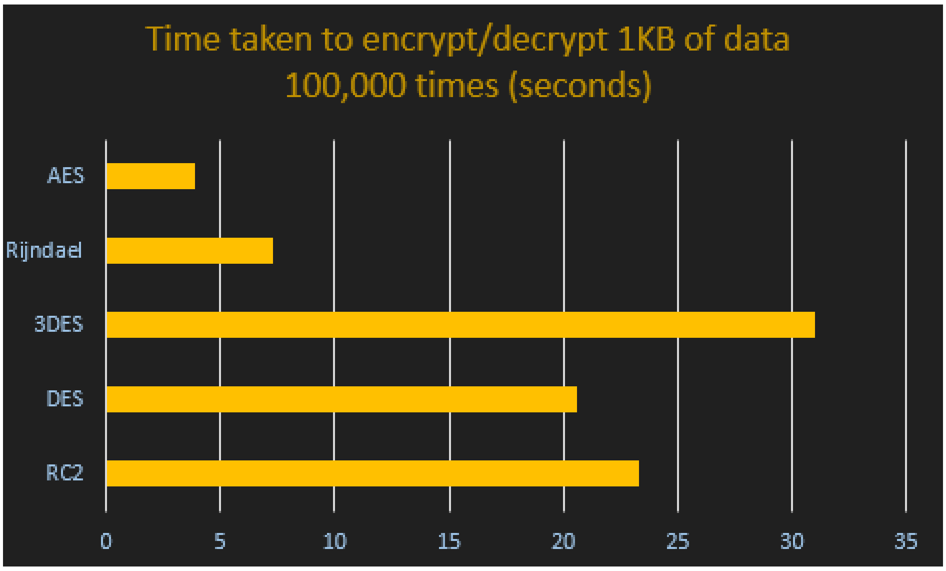
1.4.3 Which Encryption to use?

AES vs Triple DES

AES (Advanced Encryption Standard) and Triple DES(3DES) are two current standards in data encryption. AES is totally new encryption that uses that uses the substitution permutation network. Triple DES is a newer version of DES encryption and it is just DES applied three time to the information that is being encrypted.

AES uses three encryption key lengths 128,192 and 256 bits where 3DES algorithm is limited only to 56 bits however it is applied three times that is why you can have 56,112 and 168 bits , this means that you can have 3 discrete keys , or two identical keys and one discrete also you could have three identical keys. The security is reduced when reapplying the same encryption three times. This means that in 3DES the encryption of 168 bits has a reduced security comparing to 112 bits and the 112 bits has reduced security of 68 bits. AES makes it a lot harder to make the to retrieve data from identical blocks. In terms of Security between this two the absolute winner is AES which is unbreakable in practical use.

When it comes to performance between two AES is around 8 times faster than Triple DES (See Figure 1.4.3). 3DES encryption takes longer than AES encryption and also has shorter and weaker encryption keys. In addition, the time required to check all possible keys at 50 billion keys per second in AES for a 128-bit key is 5 x 1021 years, whereas 3DES with a 56 bit key would take 400 days. [1]



1.4.3Current solution to the problem.

The current solution for the problem right now is using google spread sheets where the room is booked for a certain time and can be accessed from any computer and changeable by anyone in the company who has access to the link. This problem still exists as not everyone is checking the link. The people are invited by writing a personal email to each person explaining the meeting agenda. The notes are being stored on the paper or personal laptop . The first is very unsecure as anyone can get hold of the notes as well as it is very easy to loose piece of paper with confidential information. The second option is more secure however still anyone can access the clients computer and get hold of the notes, also it is very easy to delete the document or very time consuming to find the exact notes inside the folder with many files . There are solutions as buying a software which would the company would have to pay for either annually or monthly. This is a good solution for a big company however much safer option is having a unique software with data kept on the internal server at the company premises.

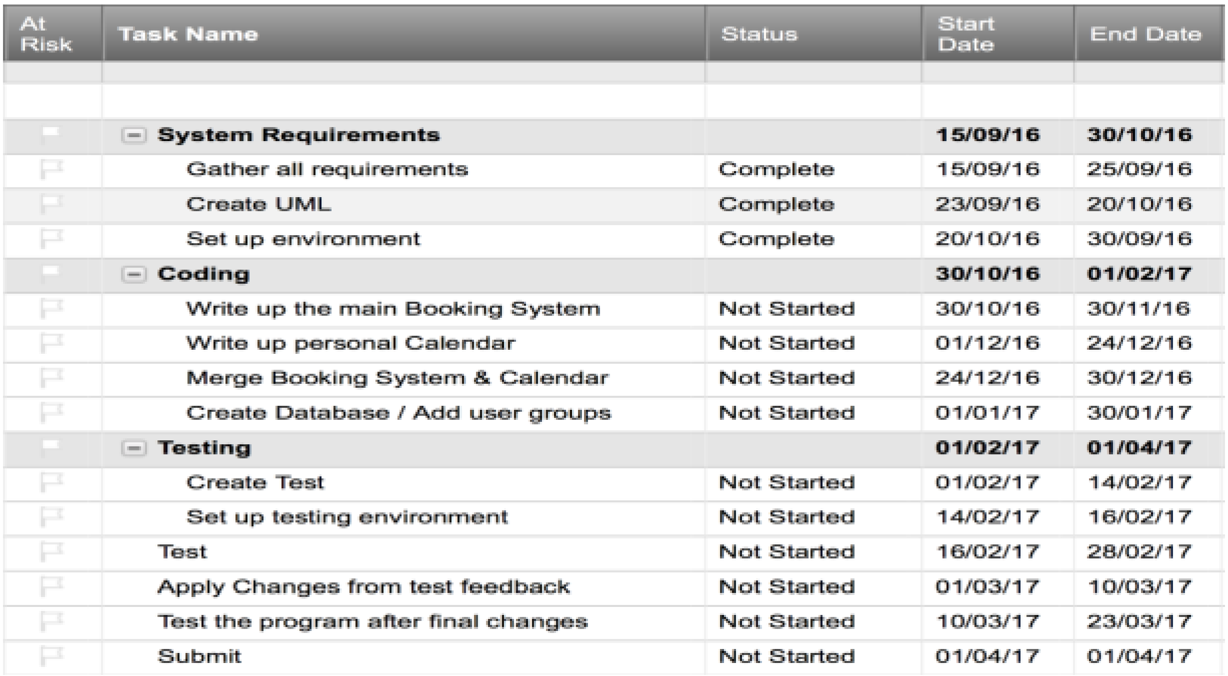
Specification

2.1 Deliverables

The first intermediate product will be creating Database using Postgres. This will contain all the users information, created events, users invitations and encrypted notes. The database will later be stored on the server inside the company premises. This development will be overlapped with the development of the admin side of application and the Gregorian Calendar in java. This will allow to add admin to the application who will be able to manage users data. The first end deliverable will be the users side of the application. This will be broken down into several intermediate products, creating new events, editing events, creating notes/reading notes, inviting/removing users from event. After three of these deliverables will be finished, I will provide dummy data for the user testing. After the software has been user testes I will receive the feedback from the co-workers and managers from World of Books who will give their opinions about the product and the changes that have to be done before the final delivery of the software.

The second end deliverable will be the final report containing documentation and screenshots of the applications explaining how the application works and the chosen ways of delivering this
software.
The initial estimated schedule of activities to produce these deliverables can be seen in figure 2.1.

Figure 2.1



At Risk	Task Name	Status	Start Date	End Date
	System Requirements		**15/09/16**	**30/10/16**
	Gather all requirements	Complete	15/09/16	25/09/16
	Create UML	Complete	23/09/16	20/10/16
	Set up environment	Complete	20/10/16	30/09/16
	Coding		**30/10/16**	**01/02/17**
	Write up the main Booking System	Not Started	30/10/16	30/11/16
	Write up personal Calendar	Not Started	01/12/16	24/12/16
	Merge Booking System & Calendar	Not Started	24/12/16	30/12/16
	Create Database / Add user groups	Not Started	01/01/17	30/01/17
	Testing		**01/02/17**	**01/04/17**
	Create Test	Not Started	01/02/17	14/02/17
	Set up testing environment	Not Started	14/02/17	16/02/17
	Test	Not Started	16/02/17	28/02/17
	Apply Changes from test feedback	Not Started	01/03/17	10/03/17
	Test the program after final changes	Not Started	10/03/17	23/03/17
	Submit	Not Started	01/04/17	01/04/17

2.2 Analysis

2.2.1Why Room booking system?

The first major point of analysis is why I have decided to create a Room Booking System & Personal Calendar rather than anything else, maybe more original. The main reason is I am working at the second biggest seller of used books in the world which doesn't have such software in place. This causes a lot of problems around the company. This made me think that I could do two things at once which are my final year project for university as well as creating a software which will resolve the issues in the company. My ambition is to get into a software development company where I could create more applications like this one and this application will be something that I can show to people and be proud of.

2.2.2 Programming Language

I have chosen Java for this project as I have been working with this language for the past three years. Java has all the features that are crucial for this software also I have had working experience with java at placement. I have used Java for many projects over the years and it makes me feel comfortable using it and visualise how the project would be written in this programming language. The other option was the C++ which is very similar to Java however I have had less experience using this language and it would make it much harder if approached problems.

The other dilemma I had to make a decision for was if to make a Java web application or desktop application this was the matter of experience, I've had much more experience in building desktop applications than web applications. Another impact on creating a desktop application was that I wanted to develop an offline version for the software where the user can access their calendar and notes when they are offline in case the server or internet goes down at the company.

2.2.3Why Postgres?

I have chosen Postgres database for the same reason I have chosen Java, this is because I have had a lot of experience writing Postgres functions at the placement. The company is mainly using Postgres database and it is the software that is placed on the company server. The Infrastructure manager at the company has advised me to use this database for the project as the software is already in place on the server and it could cause problems as the two databases would be fighting for the same resources (I/O, CPU and Network), which could become a bigger problem for ecommerce company which is focuses on the performance.

2.3 Quality Analysis

The main measure of success will be when all their requirements will be met and all software functionalities will be working. I will have to make sure all of the functionalities are tested to the finest and assures the quality of the project.

2.4 Risk Analysis

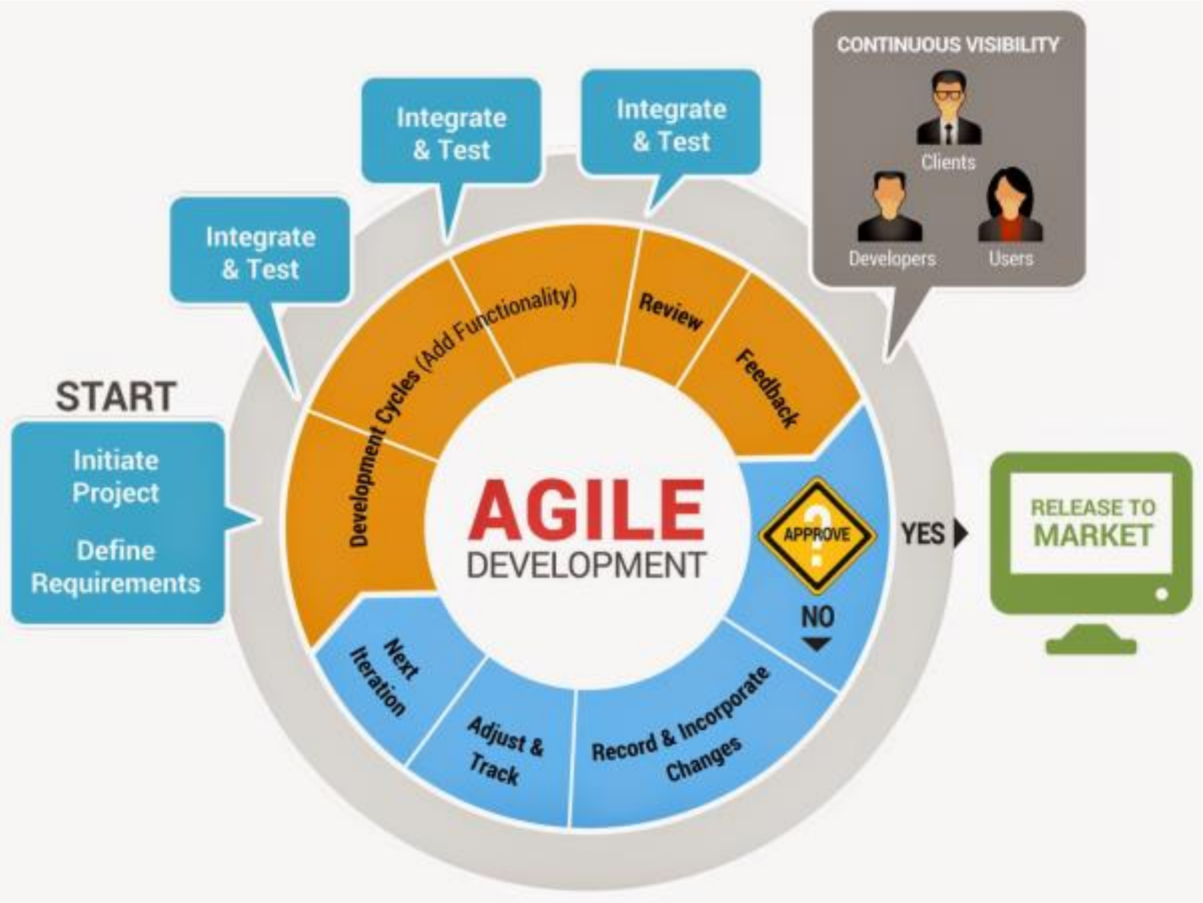
Risk	Probability (1-5)	Impact (1-5)	Mitigation	Contingency
Illness/Injury	4	3	-Reserve time for illness -Be hygienic -Eat healthy -Exercise	- Allow time for recovery - Take medicine to aid recovery
Inaccurate estimations	3	3	-Be liberal with estimations -Reserve tune fir deliverables behind schedule	-Adjust scope of project
Data loss	1	5	-User a version control system -Keep local backups	- Recover data from git
Uncommunicative stakeholder	1	3	-Ensure regular meetings with stakeholder	
Stakeholder turnover	1	4		-Get new stakeholders
Project scope too large	3	4	-Research enough to be certain in project scope -Be liberal with estimations	-Adjust scope of project
Technologies to immature/ insufficient for project	2	4	-Research technologies beforehand	-Find alternative technologies Adjust scope of project

3.1 Methodology

I have chosen the agile development methodology for this project as it has suited me and the clients, also I have had previous experience working with this methodology. Working with agile allows me to pick tasks from the backlog one at the time and progress through the cycle. At this project, it was very convenient as I would meet with the clients at World of Books twice a week where they would review the code and give me feedback for the completed tasks.

1. To Do
2. In Progress
3. Test
4. Review
5. Feedback
6. Done

-

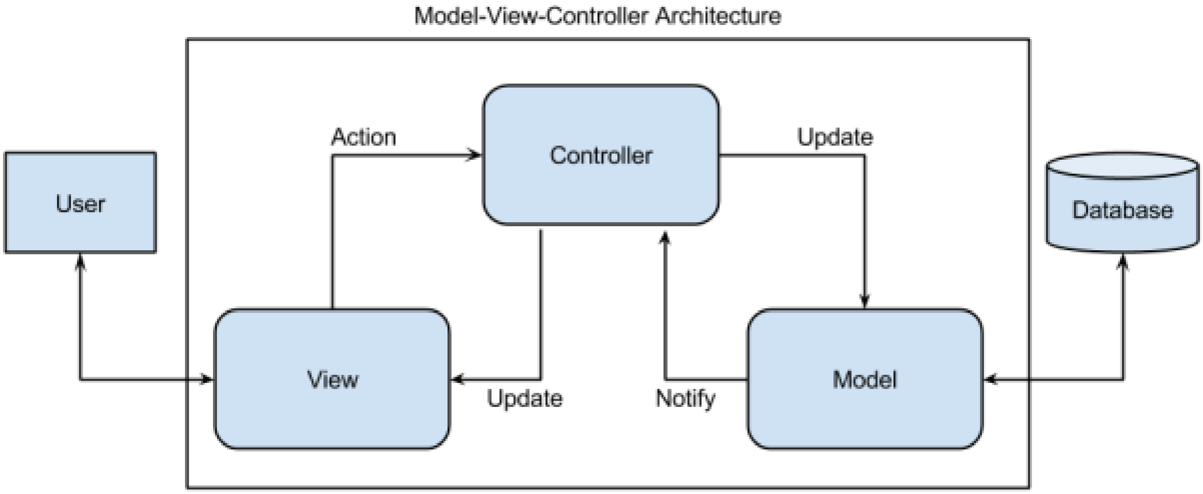


In the "To do" step of workflow are all the things that have to be done. From this section I will choose tasks one by one or maximum of two at the time. In the "In Progress" stage the code will be written to meet the requirements defined at the beginning of the project. After the code is completed than the Test writing begins, the test will check if all the functionalities of the code are working. This part is very important as it saves a lot of time in development as the errors can be found at the early stages of implementation rather than testing large chunks of code further in development or at the end. When the code and tests are completed I will allow other developers at World of Books(WOB) to check the code and review it. If the review is passed by the developers we follow to the next stage which is clients and users feedback otherwise the changes have to be implemented before passing to the next stage. At the feedback stage the management of WOB who are the clients and users give their opinion about the software if it meets their requirements and if this is what they want the software to achieve. That would be the end of the cycle if the functionality is approved otherwise the code has to be rewritten and changes have to be made.

Development

4.1 MVC Architecture

Figure 4.2 Application Architecture

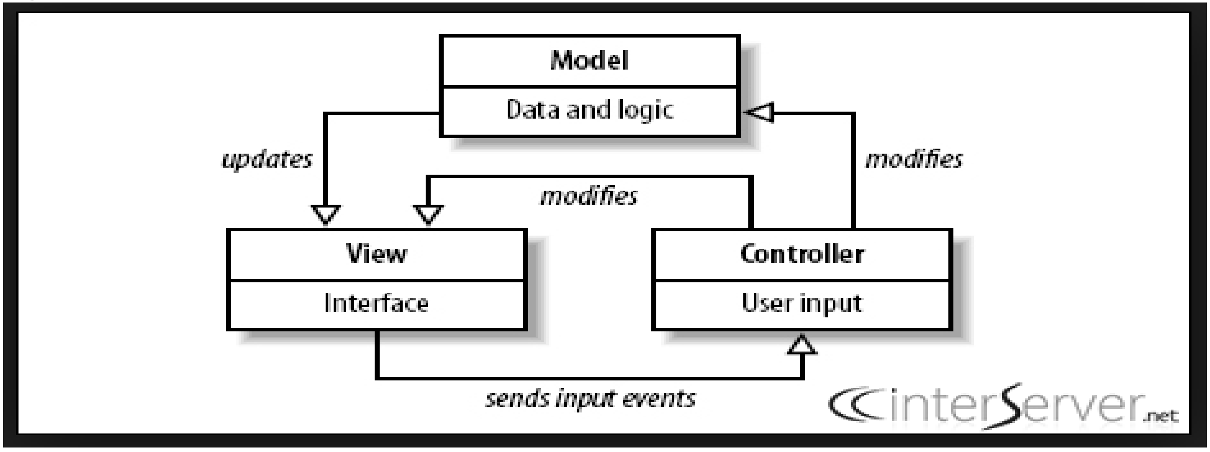


To organise the client-side application I have used the Model View Controller(MVC) Architecture. The model represents the state and business logic of the application. The view module is responsible to display data and the controller module acts as an interface between view and model. It intercepts all the requests. I chose this architecture because it is easy to maintain, easy to extend, easy to test and navigation control is centralized.

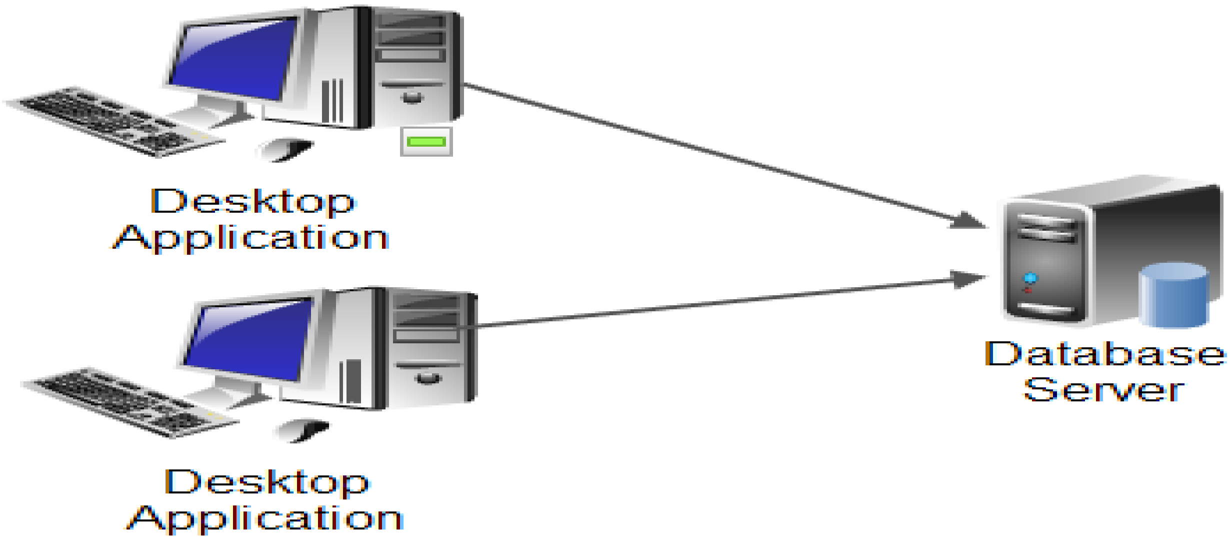
Both admin side of application as well as user side of application has been built with the MVC architecture. MVC Model returns the data without formatting: MVC pattern returns data without applying any formatting so the same components can be used and called for use with any interface.

All of the software was implemented using the MVC Architecture which connected to database on my laptop for testing.

Figure 4.2.2:



4.1.1 Client – Server



4.2.1 UML

4.2.3 Authentication

To add basic authentication to the application, I have created a login panel which involve inputting their username and password. When the user is created by admin, the username and password are chosen by the admin. The password will then be encrypted and passed to the database. When the user Inputs his username and password the username is crosschecked with the username in database and the password is then decrypted and crosschecked with the one users input. If the username and password are checked and are correct the user is granted to the main user panel otherwise the error comes up. There is no forgotten password option, if the user forgets their password they have to contact the administrator. If the password was not hash encrypted anyone with some decent skills could grant entry to the database and retrieve all the passwords.

4.3 Other Technologies

4.3.1 Git & Gitkraken

Through the whole development process, I have been using GitHub repository where I have been committing my change regularly. I had previous experience of using git as I've been on the placement where everything had to be done through version control system. I've been using Gitkraken to push and pull files. Gitkraken provides the GUI for git and makes it a lot easier to see where the changes have been made and to select the files you would like to commit. Gitkraken also provides the option to see what lines of code have been changed in a file and allows you to discard or commit single line. Git is a great habit to learn as it allows you to track your log and prevent the data loss or overwriting your files which can be catastrophically. This can happen when you use USB stick, google drive, drop box or a hard drive. Git allows you to easily switch between platforms, it was very useful for me as I have been developing at work, home and at the university. After I've finished my placement I have kept using git for my university projects. Maybe someday I will have contributors to this project who will try to develop this software further.

4.3.2 Window Builder

I have been using Eclipse plug in which is a Java Window builder. This plug in allows you to create graphical interface for the project. This software was very helpful as each class had a view and it would be very difficult using just code to lay out every component. Window Builder saves a lot of time and provides a rich API for creating UI designers. This software lets you easily switch between code and design. I've been creating GUI by writing code for every single component and it has taken me a lot of time and made me frustrated. The plug in has a lot of features as:

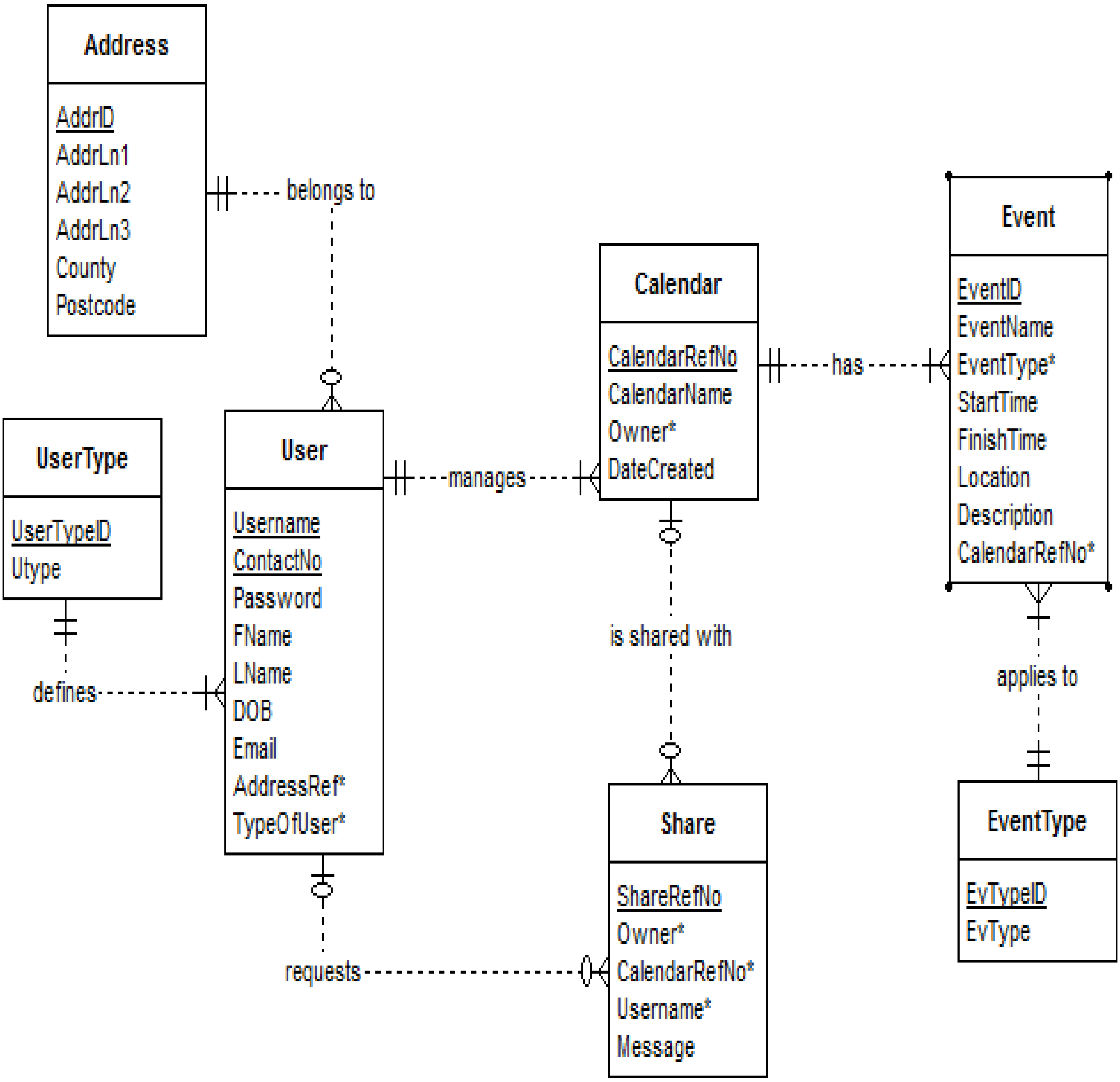
- **Bi-directional Code Generation –** read and write almost any format and reverse-engineer most hand-written code
- **Internationalization (i18n) / Localization –** externalize component strings, create and manage resource bundles.
- **Custom Composites & Panels** – create custom, reusable components.
- **Factories** – create custom factory classes and methods.
- **Visual Inheritance** – create visual component hierarchies.
- **Event Handling** – add event handlers to your components.
- **Menu Editing** – visually create and edit menu bars, menu items and popup menus.
- **Morphing** – convert one component type into another. [2]

4.2.2 ERD Diagram

Entity	Attribute	Data Type	Allow Nulls	Description
User	Username	Varchar (8)	N	This is 8 long as it will be a combination of letters taken from the user's name and sequential numbers.
	Password	Varchar (15)	N	This will include letters and numbers. This could be forced to 15 long or somewhere close to it.
	FName	Varchar (30)	N	The user's first name.
	LName	Varchar (30)	N	The user's last name.
	DOB	Date	N	Need to know when the user is born for events like their birthday (could be automatically created into an event.
	ContactNo	Varchar (12)	N	The user's phone number.
	Email	Varchar(320)	N	The user's email. This is important as shares and invitations require emails to be sent to and from users. 64 characters for the username, 1 character for the "@" symbol and 255 characters for the domain name.
	AddressRef*	Int	N	The foreign key for the user's address.
	TypeOfUser*	SmallInt	N	The foreign key for the type of user.
Address	AddrID	Int	N	The primary key for the different user addresses. This must be unique.
	AddrLn1	Varchar (50)	N	The first line of the user's address.
	AddrLn2	Varchar (50)	N	The second line of the user's address.
	AddrLn3	Varchar (50)	Y	This can allow nulls as not every address has three lines in them but the option is still available to users.
	County	Varchar (50)	N	The user's county.

	Postcode	Varchar (8)	N	The user's postcode.
UserType	UserTypeID	SmallInt	N	The primary key for the type of user. This doesn't have to be so large so small Int suits his attribute better than int.
	UType	Varchar (30)	N	The different types of users i.e. Regular/Administrator etc.
Calendar	CalendarRefNo	Int	N	The primary key for the calendar.
	CalendarName	Varchar (50)	N	The calendar's name.
	Owner*	Varchar (8)	N	The username of the user who owns the calendar. The foreign key from the user table.
	DateCreated	Date	N	The date when the calendar was created.
Share	ShareRefNo	Int	N	The primary key and unique identifier for each share created.
	Owner*	Varchar (8)	N	The owner of the calendar being shared.
	CalendarRefNo*	Int	N	The calendar's reference number which is being shared.
	Username*	Varchar (8)	N	The user who is requesting the share.
	Message	Varchar (100)	Y	The message from the requesting user to the calendar owner.
Event	EventID	Int	N	The primary key for each event created.
	CalendarRefNo*	Int	N	The calendar that the event belongs to.
	EventName	Varchar (50)	N	The name of the event set by the user who created the event.
	EventType*	SmallInt	N	The foreign key for the event type.
	StartTime	Time	N	The start time for the event.
	FinishTime	Time	N	The finish time for the event.
	Location	Varchar(50)	Y	The location where the event will be held. Some events may not have a location yet.
	Description	Varchar(100)	Y	The description of the event. Some events may not need a description.

EventType	EvTypeID	SmallInt	N	The primary key for the different types of events.
	EvType	Varchar(30)	N	The different types of events the users can select from.



Testing

5.2 Automated Testing

I have been unable to perform automated testing because of lack of frameworks for the Java desktop application. Sikuli is one of the frameworks available for Java desktop application testing . I have used Sikuli only for some of the functionalities as this involved a lot of work in setting up test by cutting out the buttons. However it was very helpful when making changes to the database where all the data was removed and the test would perform checks on the creating new user , changing the user password , changing user details and creating new events. Using automated test definitely saved a lot of time and made it easier to test all functionalities that had to be repeated over the course of development.

5.1 Manual Testing

I have been practicing manual testing through the whole project to ensure that the application is error free and it is working in conformance to the specified functional requirements. This testing checks the quality of the systems and delivers bug free product to the customers.

I had to perform Manual testing as 100% of automated testing can not be performed. Due to lack of a unit testing framework for java desktop application GUI, I was unable to write unit or integration tests. To overcome this problem I’ve been testing new functionality every time it was added. I was using manual black box testing with different possible cases to check if the specific functionality is working correctly and returns positive results. I’ve been testing the software on different platforms and also different users to check if the software is working properly. The users gave back their results and opinions about the functionality. If the everything was working as it should then I would move on to the next task however if there were changes to be made I would perform the changes and perform the test again.

Review & Reflection

7.1 Background Research

The background research that was investigated prior to the development of the project has definitely helped me understand many aspects of databases, programming languages and the most important encryption. I never had a chance looking into encryption and this project has definitely helped me achieve this . I feel as the database research made me understand the real difference between the MySQL and PostgreSQL. The research got me more involved into looking at C++ . It has shown what the language is really useful and what features it has and which features it lacks compared to Java. The encryption detail made me realize that AES encryption is the most efficient algorithm out there and that it should be used when handling confidential data.

7.2 Specification

7.3 Methodology

Through out the project's timeline I was able to keep to Agile methodology approach. However I have met some obstacles and difficulties across the development such as management or the developers couldn't find time to review the code so I could progress further with implementing next functionalities. Another difficulty was writing test codes as it was my first time writing tests for my own code , however after doing some research I was slowly progressing through tasks.

7.4 Limitations & Improvements of the application

7.4.1 Desktop application

The software currently is a Java desktop application . The choice was made at beginning of the project to create a desktop application . Working through the project I've realized I would be better of creating a web application that would either be stored on the local server or be hosted outside the company so the users could use it at home as well. The graphic interface is one of the limitations when creating a desktop application , where in a web application there are many frameworks that can be used for e.g. Foundation , YAML or the most popular which is Bootstrap.

Another disadvantage of using desktop applications is that the update to the applications must be installed by the admin directly on the users computer and may require system updates in order to work. Updating web application means you have to update the web application once and the changes appear on the users screen for e.g. changing the user interface .Creating web application would allow users to use the application on their mobile phones by including css styling for mobile phones where in the desktop application is not possible and would mean that there would have to be a separate mobile application to be developed.

I could have chosen the web application over the desktop application however the desktop application suits the company needs and has contains all functionalities that were collected in the requirements stage of the project.

7.5 Conclusion

References

[1] A Comparison of the 3DES and AES Encryption Standards - Noura Aleisa

[2] https://examples.javacodegeeks.com/desktop-java/ide/eclipse/eclipse-window-builder-tutorial-gui-creation/