MONASH University

Detecting heart disease: Project proposal document

Abrar Hamzah | 28551494

Tom Orlando | 28009592

Jian Tan | 28384695

Team 6 | 8370 words

FIT3163 | Mary Lim

Contents

Introduction

In today's society, cardiovascular diseases are the main cause of death around the world. Coronary artery disease (CAD) leads to restricted blood flow as a result of blocked arteries from cholesterol and fatty deposits. The aim of this project is to design and implement various data mining and predictive modelling methods, in an attempt to predict the occurrence of CAD in patients, based on a particular set of parameters. This will be done through two key workstreams next semester: 1) a predictive model and 2) an interactive user interface.

This document aims to build a complete and extensive project proposal for the use case described above. The proposal will include a literature review, to update readers on recent advancements in predictive machine learning methods, a project management plan describing the relevant project processes, as well as segments discussing external design, methodology and test planning for the project.

Literature review

Introduction & research context

The recent surge of global data has empowered machine learning - a field of computer science that uses statistical techniques to make data-driven predictions - to provide organisations with a method to make better decisions with minimal human intervention. Following considerable success on a broad range of predictive efforts in the medical space, machine learning classification methods are attracting visible interest from medical researchers and clinicians globally. The following literature review explores various classification methods used to predict illness in patients, particularly heart disease, discusses these methods with the aim of understanding how they are being used and interpreted in various studies, and concludes there are a subset of these methods which tend to provide more precise and accurate predictions, based on the particular use case of detecting heart disease.

Extensive research on various machine learning classification methods in the last decade has been undertaken to complete this review. In gathering the literature underpinning this work, a three tier methodology has been undertaken as a way to efficiently organise research analysed. The first tier includes all literature with a broader focus of using classification methods to make predictions generally, with no specific use case. The second tier encompasses research articles with a focus on using classification methods to make predictions in the healthcare and medical space specifically, whilst the third tier looks at studies with a particular focus classification methods used to predict heart disease in patients. This methodology has provided a broad view of the existing research landscape, whilst enabling control over the acuity and specificity of the research scope.

The research gathered and analysed, has been used to answer a set of questions into the use of classification methods to predict heart disease in patients. Specifically, there are two main questions the literature review addresses 1) what are the fundamental classification methods used in machine learning over the last decade? And 2) what are the leading classification methods that should be used to accurately predict heart disease in patients? In answering these two questions, this literature review will provide guidance as to what methods are determined to be the most effective and should therefore be used in the proposed project.

Analysis of fundamental classification methods

Logistic regression

Logistic regression is a classic classification method, which has been widely used in the field of medical sciences to underpin various predictive and diagnostic models. Reddy & Delen's (2018) research utilises deep learning methods to predict the re-hospitalisation of patients with lupus, by extracting temporal relationships in clinical data. Prediction results from new learning methods such as LSTM neural networks (LSTM) are evaluated and compared with penalised logistic regression and artificial neural networks. In their penalised logistic regression model, Reddy & Delen employ a hybrid regularisation method called elastic net to minimise variance in the model and to mitigate the risks of under/overfitting. The elastic net method is a combination of the ridge and lasso methods which are commonly used in penalised regression, and often combined with cross validation (CV) to select a tuning parameter λ. Results of the study show LSTM has a significantly better performance compared to logistic regression.

Easton, Stephens and Angelova's study (2014), illustrates the benefits of a data mining approach with the aim of limiting some of the inherent bias in the hypothesis assumptions found in traditional clinical data analysis, and in doing so, accurately predicting stroke mortality. This was done using a Naive Bayes model, compared against other classification methods which include logistic regression and decision trees. The logistic regression model was created in a stepwise fashion in order to find the best fitting model based on the Akaike Information Criterion (AIC), with the study noting that it is recommended to have a large number of cases per independent variable. Results show the optimised logistic regression performs well in sensitivity but comes at a greater cost of specificity, implying logistic regression cannot reach its asymptotic error rate, which is typically lower than naive Bayes for larger data sets. Another study (Dag, Oztekin, Yucel, Bulur & Megahed, 2017) also uses a stepwise approach to logistic regression with the aim of predicting a patient's graft survival following a heart transplant surgery. Again the regression model is contrasted to other classification algorithms, however results show that logistic regression, combined with a synthetic minority sampling technique (SMOTE), achieves the best classification accuracy for the outcome predictions.

Finally, a study by Prashanth, Roy, Mandal and Ghosh (2014) makes the use of logistic regression and support vector machines (SVM) in developing automatic classification and prediction / prognostic models for detecting early Parkinson's Disease. In their regression model, Prashanth et al. (2014) include interaction terms between variables (specifically the

product of two independent variables) as a way to construct a better fitting model, as they deal with limited features in the data. The performance of the logistic regression model was subsequently evaluated using the four step framework carried out in Peng, Lee & Ingersoll's 2002 study, and results show logistic regression, though its prediction accuracy is relatively high, did not perform as well as SVMs.

In analysing recent research, it can be seen that logistic regression appears in various studies and is one of the main methods used by experts in predicting illnesses in patients. It is apparent however, that logistic regression, being a more traditional classification method, is often used as a benchmark to contrast the performance of other more novel methods of prediction, rather than a leading classification method. Logistic regression also seems to be less accurate than other models in most studies (excluding Dag et al.), likely due to logistic regression models being discriminative models for classification that produce linear boundaries, and are not as flexible as non-linear models such as SVMs or decision trees. The review also shows there are a plethora of ways in which to build and evaluate linear regression models, which include penalised regression, stepwise methods and interactions for building models and sensitivity analysis, area-under-the-curve (AUC), and model evaluation frameworks (Peng et al.) for the evaluation. The review also shows there are gaps in the research landscape, with little studies focusing only on logistic regression to predict heart disease in patients. Overall, the review shows logistic regression to be a widely occurring method of classification and should be used in the project as a benchmark to evaluate other, more complex methods such as ANNs, decision trees and SVMs.

Decision trees

Decision trees are one of the most widely used and effective predictive modelling approaches in data mining and machine learning to date, and have had great results in the medical space. A study by Guo, Zhang and Yu (2020) showed that ANNs are widely regarded as the best way to predict diseases such as encephalopathy and heart disease. However, various studies tend to contradict Guo et al., proposing other methods (including decision trees) as more efficient and accurate than ANN in predicting heart disease. In a comparison between ANNs, naïve Bayes and decision tree algorithms, Ansari & Soni (2011) found the decision tree was the most accurate of the three in coronary artery disease (CAD) prediction. Tayefi, Tajfard & Ghayour-Mobarhan's study (2017) also states that decision trees used as data mining models to extract hidden knowledge from large databases, are more efficient than previously discussed counterparts. Some of the reasons include interpretability of the model, and the high prediction accuracy (94%) found in classifying patients with CAD.

Ghiasi, Zendehboudi & Mohsenipour (2020) systematically studied CAD classification based on Z-Alizadeh Sani dataset and have formulated their analysis based on a decision tree learning algorithm called classification and regression tree (CART) for simple and reliable diagnosis of CAD. CART is a non-parametric machine learning method, which is deemed as one of the most successful techniques used in classification tasks and regression analysis. Being a non-parametric, allows the CART method to freely learn any form of mapping function from the training data samples used and does not depend on or belong to a specific distribution type. As a result, the algorithm is flexible and powerful. However, it requires

more training data sets than parametric methods such as naive Bayes and linear discriminant analysis. It is worth mentioning that the outliers in the input parameters will not significantly affect the performance of CART, however unbalanced classes may result in under-fitted trees.

In analysing recent research, it can be seen decision trees are widely used instruments in the medical space and have proven highly effective across various use cases. The interpretability of decision trees often discussed throughout research also provides a fundamental benefit to patients which need to understand the diagnosis, contrasting "black box" approaches such as ANNs and SVMs. There are minimal gaps seen in the research, as decision trees seem to have been very heavily used in the medical space (and classification space in general), however recent developments in decision tree approaches such as the CART method also provide an interesting base for building more accurate and sensitive predictive models. Overall the review shows decision trees form a fundamental part of classification exercises both in the medical space and generally, and will be implemented in the proposed project.

Support vector machines

Research conducted as part of the literature review shows SVM methods have been frequently used in feature selection and classification exercises in the last decade. Alizadehsani (2013) states that the "Weight by SVM" method has proved to be effective for predicting CAD, by leveraging various software tools such as RapidMiner to better understand the importance of predictors and which combinations make for the optimal model. Zangooei,Mohammad & Jalili (2012) also applied Weight by SVM to both select features and build a complete SVM model which presented comparable or better prediction results than neural networks in bioinformatics applications. The SVM model passes a new parallel multi-class (PMC) method, parallel hierarchical grid search (PHGS), cross-validation (CV) technology and weighted kernel fusion (WKF), in an attempt to build a more accurate model. Alizadehsani, Zangooei & Nahavandi (2016) also holds that different well-known feature selection methods, such as filter method and wrapper method are less effective than Weight by SVM and that average information gain and combined information gain also have good accuracy.

Guo et al. (2020) analyse the data set includes 99 CAD patients and 94 patients with normal coronary arteries. In disease prediction, the correct rate is higher and there are fewer errors, resulting in the SVM model performing better than other models with the highest accuracy. This insight has been confirmed in another study by Babaoglu, Fındık & Bayrak (2010), which illustrates that Support vector machine (SVM) CAD diagnosis accuracy rate reached 81.46%, higher than all other models analysed. The optimal SVM model showed using principal component analysis to reduce the data set to 18 features is more accurate than the optimal support vector machine model using the entire 23 feature data set and that due to its powerful binary classification, it can be used in many applications. Alizadehsani et al. (2016) also state that some supervision methods are used, including Naive Bayes, Decision Tree (C4.5), Association Classifier and SVM. In the experimental results, SVM showed the best performance with an accuracy of 90.9%.

Throughout the research analysed, it is clear SVMs are deemed to be powerful tools in both feature selection and classification. Most of the studies seem to rank SVM as either one of or the most accurate model in their set of classifiers, often outranking the likes of naive Bayes and logistic regression models. Different software applications which further enable the use and configuration of SVMs such as RapidMiner, also provide a catalyst for further application and implementation of SVMs in the proposed project. There are again minimal gaps seen in the research of SVMs as there have been multiple studies using the model to predict CAD. Overall this review depicts SVMs as essential tools to use in classification, especially in the prediction of heart disease and CAD, making it an attractive method to use in our project.

Naive Bayes

Naïve Bayes algorithm is a well-known simple algorithm for classification, and it is used frequently in research to classify and predict. One study by Zarndt (1995) shows that NB performs worse than other algorithms which led to further research into Zardnt's findings, (Jamain & Hand, 2005). The claim by Zarndt was disputed by Jamain and Hand (2005), with the study suggesting that the naive Bayes methods used in the previous research were incorrect. The authors found an anomaly that the implementation used a Bayesian tree method which is different to naive Bayes classifier. The statement from Jamain and Hand was then seconded by Al-Aidaroos, Bakar and Othman (2012), that different naive Bayes were used, and the different datasets types used were the cause of the poor performance of NB.

In more recent studies, Naïve Bayes is reported to have a high percentage of accuracy on health diagnosis. It is proven by Al-Aidaroos et al. (2012). Their aim of the research was to compare different algorithms to naïve Bayes on different medical problem cases. The research used the WEKA tool which contains methods for data pre-processing and classification. Furthermore, WEKA was used to compare the performance of Naïve Bayes against other algorithms. The outcome of the research shows that Naïve Bayes outperformed the other algorithms in most cases. In fact, Naïve Bayes outperformed 8 out of 15 cases such as breast cancer, dermatology, and primary tumour, to name a few. This backups the statement of Naive Bayes classifier is known to perform well in medical diagnosis.

As for predicting heart disease, Venkatalakshmi & Shivsankar (2014) stated that Naïve Bayes has a higher accuracy then decision tree with accuracies of 85.03 and 84.01, respectively. However, according to Venkatalakshmi & Shivsankar (2014), the most important disadvantage is its strong feature independence assumption". It shows that Naïve Bayes assumes features that are used and not used are irrelevant to each other. Naïve Bayes could achieve better results when the problem and dataset is controlled properly, and better accuracy and efficiency could be achieved in large datasets (Saritas & Yasar, 2019). The accuracy of naïve Bayes classifiers can also be improved by using more sophisticated naïve Bayes, selective naïve Bayes classifiers and relaxed Naive Bayes (Nikhar & Karandikar, 2016).

In conclusion, Naïve Bayes classifier is good enough to predict medical diagnosis including heart disease however, it would be best to improve the accuracy and reliability by using

other tools or by hybridisation and by controlling the data and problem closely, as stated in Al-Aidaroos et al. (2012). Overall the research in the review shows that though naive Bayes models can work well, their simplistic and classical approach, coupled with their fundamental assumption of feature independence (which is often not the case in analysing CAD data) tends to rank lower on our list of methods to use in the project.

Artificial neural networks

Neural networks are models that are known to perform like a human brain in the learning process, by learning through examples and patterns in the data (Hannan et al., 2010). These work by inputting the next layer with the output of the previous layer. There are several types of neural networks, with the most frequently used being Multilayer Perceptron (MLP) which has a structure of three layer; input, hidden and output layer. This algorithm is best to find a hidden pattern in the data and it is very flexible and powerful (Hannan et al., 2010).

The performance and efficiency of neural networks was recognised globally. One study shows that the accuracy compared to decision tree, Naïve Bayes, and logistic regression on 15 medical cases is not vastly different from one another (Al-Aidaroos et al., 2012). However, the study implies that it is not the best algorithm since it only outperforms those algorithms once out of the 15 cases. In another study by J. Soni, Ansari, Sharma, and S. Soni (2011) outputted similar results. Neural network's accuracy just barely falls under the decision tree and naïve Bayes. Decision tree is 89%, naïve Bayes with 86.53% and ANN with 85.53% accuracy. A study on predicting heart disease using ANN was also conducted. The result was similar, with ANN not performing as well as other models. These studies have a common method which was done using MLP neural networks with Back-propagation. Back-propagation is done by comparing the first output of the system to the real output and the algorithm adjusts the calculation to achieve minimal error (Saritas & Yasar, 2019).

According to Kahramanli & Allahverdi (2008), back-propagation alone does not perform well. Therefore, their study was around hybridisation of ANN. They used a fuzzy neural network trained using the back-propagation algorithm. They also used k-Fold cross validation to test the results. According to the result in the paper, the hybrid neural network has the second highest accuracy of 86.8% on predicting Cleveland heart disease and the highest on predicting diabetes with an accuracy of 84.2% compared to previous research, as cited in Kahramanli & Allahverdi (2008). Another study regarding asthma also shows that hybridisation of neural networks achieved higher performance. As Aneja and Lal (2014) stated in their research, combining back-propagation neural networks and naïve Bayes classifiers achieved better accuracy. The accuracy for naïve Bayes itself is 88% and neural networks alone achieved 85% while the combination output reached 93.5%. The method they used incorporates additional knowledge into naïve Bayes through neural networks. However, recent study shows that these hybrid approaches have a limitation on automatic calculation of weights of the attributes (Raja & Asghar, 2020).

It can be seen throughout the research that there are several types of artificial neural networks and different approaches achieved different results. Notably, various studies show that ANNs implemented on their own often fail to outperform more classical and simple

approaches and that hybrid methods, which can take various forms, are often required to get competitive results. Based on the wide range of potential combinations and methods in implementing neural networks, there are still gaps in the research and ways to produce better methods to predict CAD with neural networks. Overall, though neural networks can work underperform in certain circumstances, this review of literature has shown with the right methods in place such as backpropagation and hybridisation, some ANNs can prove to be powerful tools and should be highly considered throughout this project.

Evaluation of literature & conclusion

This literature review has explored various classification methods used to predict illness in patients and has extensively discussed such methods, analysing how they are being used and interpreted in various studies. The methods discussed - which included logistic regression, naive Bayes, decision trees, SVMs and ANNs - answered the first question posed at the start of this review, showing what the fundamental classification methods used in machine learning over the last decade. In further analysing these methods, we were able to find some methods with more attractive factors including accuracy, interpretability and simplicity, whilst others lacked more. This analysis has allowed us to answer the second question posed at the start of this review, in which three fundamental methods to use in predicting heart disease will be decision trees will be used, SVMs and ANNs, whilst logistic regression and naive Bayes classifiers will act as performance benchmarks to contrast and compare the three models. Generally, more research will be required to gain a better understanding of the relationships between these methods and how to best implement, test and distribute such models in a working product to predict heart disease, where accuracy and reliable implementation can be the difference between a patient's life and death

Project management plan

Project overview

The aim of this project is to design and implement various data mining and predictive modelling methods, in an attempt to predict the occurrence of coronary artery disease (CAD) in patients, based on a particular set of parameters.

The final implementation of the product will consist of two segments: 1) predictive model which uses machine learning and statistical methods 2) an interactive and responsive user interface in which the model is integrated. These two segments combined aim to provide an accurate and reliable tool - for both medical practitioners and patients alike - to predict a potential CAD based on the patient's unique traits. This sort of tool allows for a cheap diagnosis (or a suggestion to get tested), with no appointments and operations required, which will prove beneficial to many patients around the world which have limited access or resources to do so.

Project scope

Project deliverables

Project deliverables will include everything that we've produced as a team for the project. These can be categorised into two succinct groups which include 1) user deliverables and 2) administrative deliverables. User deliverables will include everything that has to do with the final product and that users will be able to see or use in some way. This will include the web application itself, pages with information on CAD included in the website, predictive functionalities (i.e. underlying predictive models) and accurate prediction outputs. On the administrative side, deliverables include all of the work that has allowed us to initially build the product and manage the project. These include a project charter, project schedule, literature review, team contract, WBS, scope statement, time and risk management tools, status reports, project proposal. Understanding these deliverables will be crucial in implementing and building the project as a whole.

Product characteristics and requirements

After determining all deliverables, we need to record all the needs of the project. Product requirements define the values and purpose of the product, the characteristics of the final output and the necessary functions that the final deliverable must have in order to satisfy user needs. These include:

- the ability to compare different feature selection algorithms
- the ability to compare different classification algorithms
- the ability to accurately predict whether a patient has CAD based on their unique characteristics
- Ability to upload their personal data to the database through the UI
- Provide a clear and direct dashboard on UI for users
- Users can easily operate and use comprehensive functions to get their own results they want
- When an operation error occurs, an exception report is submitted and tips are given to correct

Product user acceptance criteria

Product user acceptance criteria allows us to determine if we have achieved user objectives in the final product. This will include criteria from the user deliverables described above, ranging from the look and user experience of the website to the handling of errors and general functionality of the site. These have been described in more detail in the table below, which will be used throughout the project to understand if we have achieved specific user criteria.

Criterion	High quality	Expected quality	Low quality	**Score**
Accurate and timely prediction				
Efficient and intuitive design				
Ability to add all characteristics				
Informative website				
Ability to use website functions for desired results				
Website errors are minimal and dealt with efficiently				

Project organisation

Process model

The process model we will use throughout the project is the iterative process. This is an approach which allows feedback for unfinished work to improve and modify specific areas on an iterative and ongoing basis. This method stresses an approach of building "imperfect deliverables" first, and through feedback from stakeholders, gradually tweak and fix features until an adequate level is reached.

This method has been chosen based on the nature of this project being partly an inquisitive study to find algorithmic methods to best predict CAD. As there is no clear cut definition for the best model currently, there will be a lot of model testing with various combinations and feedback required and an iterative approach will prove most useful. This will be similar with the website app deliverable, where an initial product will be built and through feedback and testing, will be tweaked to ensure the best performance and that all user acceptance criteria are adequately solved.

Project responsibilities

As shown in the WBS and Gantt chart in *Appendix 3* , there will be three stages. The roles needed in the first stage is data scientist. Their main work is to pre-process the data and create a warehouse where all data will be stored and updated after the user input their data. In addition, data scientists should create the model needed to predict heart disease and review each other's code and test different algorithms. This will ensure that all algorithms run properly and will produce correct output or expected results on predicting CAD. Since, we are working on a group of three all of us will contribute equally. Jian will start with the data pre-processing while monitored by Abrar and Tom. For feature selection, Abrar will do

Pearson correlation and stepwise algorithm, Tom will be responsible for LASSO and metaheuristic algorithm.

Our goal of the project is to create an interactive website. Therefore, the nature of the second stage is around user interface, interactivity, and security system. It is the responsibility of web designer and IT security however, only Tom and Jian that had some experience on making websites and UI while Abrar will help them in any task while doing research and online courses. It is expected that the webpage will be functional in a week because we will need more time for our security system. Lastly, the third phase will include reviewing our project and some testing and monitoring. This stage will be done by the three of us from week 11 to 12 to make sure our software runs smoothly and correctly just before the submission date.

Roles	Responsibility
Data Scientist	• Review each other's code • Finish the task as soon as possible • Research on each algorithm and justify decision making
Web Designer	• Create website that is appealing to users • Focus on interactivity • Communicate with IT security on functionality and Data Scientist on implementing the data analysis
IT security	• Ensure no data input error • Create the web as secure as possible while maintaining simplicity
Organiser	• Make sure all files and codes are stored • Update and note progress of project • Ensure activities are completed by the given milestone date. • Take care of meetings, schedule, and other roles

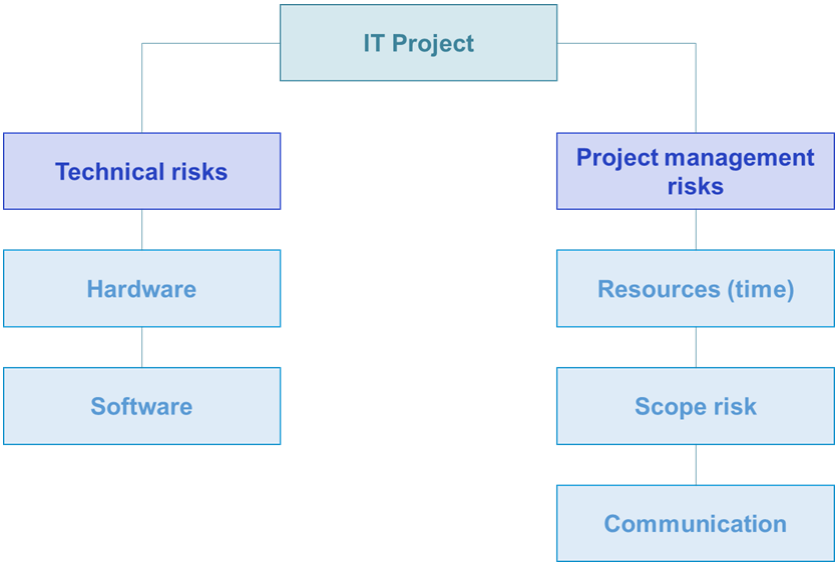
Management process

Risk management

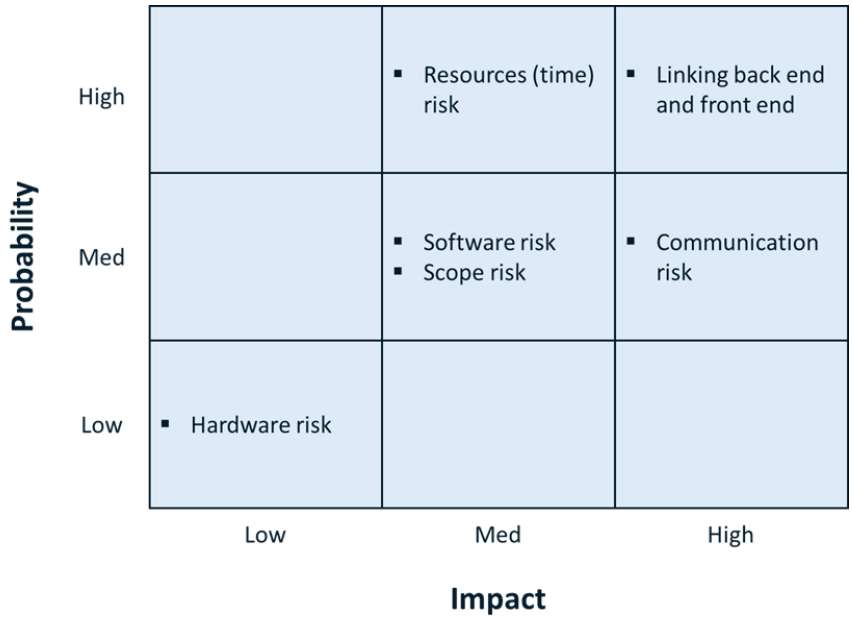
Project risk management is the process of identifying, analysing, and responding to risk in the best interests of meeting project objectives, throughout the life of a project (Schwalbe, 2015). It is important to understand there are both positive and negative risks commonly arising throughout projects, both of which carry their own effects on meeting project objectives. This section aims to describe the risk management process for the project which will include identifying, analysing / prioritising and managing / controlling project risks.

The following *risk breakdown structure* (RBS) aims to identify the hierarchy of risk categories within the project, allowing us to identify and categorise risks. These risks are initially identified through a *brainstorming* session, as a way to explore the potential risk landscape with a consensus amongst group members. Elements of SWOT analysis, to understand key strengths and weaknesses within the group, also plays a part in formulating

the RBS. It should be noted the risk identification process is iterative and will be undertaken multiple times throughout the project lifecycle.



Once risks have been identified, further steps are taken to analyse and prioritise risks. This is initially done through the use of a *probability / impact matrix* to prioritise the most important risks and gauge the potential cost they might bring to the project (see below). Subsequently, a *Top Ten Item Tracking* method - built leveraging the impact matrix - is used to maintain an awareness of prioritised risks throughout the life of the project. At each iteration, the team will list the current ranking, previous ranking, number of times the risk has appeared and a summary of the progress made in resolving the risk item. Note an in-depth risk register is also included in *Appendix 1*.



Once risks are identified and analysed, it is important to effectively plan risk both negative and positive risk responses, as well as understand how the team will control these risks throughout the life of the project. For negative risks we are using the *TARA* approach, in which risk response strategies are split between four key categories: 1) risk transference 2) risk avoidance 3) risk mitigation and 4) risk acceptance. A similar approach is also used for positive risk. A risk management and response approach like TARA, provides the team with different strategic "levers" to use in case various risks arise, in which each lever is suitable for different types of risk, depending on the situation at hand. Each time a risk response is implemented, the team will regroup to discuss residual and secondary risks through another brainstorming session as a way to recalibrate existing risks identified and analysed and plan accordingly.

The complete risk management process described above should be an iterative and dynamic process, with constant monitoring and ensuring that risk awareness is communicated within the team on an ongoing basis, through regular catch ups and discussions. The processing and planning of such communication and reporting will be discussed in the following section.

Communication and reporting

Effective communication within teams is fundamental to project success and can be one of the main reasons for increased risks and failure in a project. As a way to promote communication as a priority throughout the project, we have taken a three part method to effectively implement communication processes and expectations in the team, which includes planning, managing and controlling communications.

In planning communication management, it is important to focus on both the group and individual's preferences for communication. This entails forming communication methods which fit each individual's preferred ways of working, ensuring no extra risks or issues are created in doing so. After multiple discussions, the stakeholders of our project have found a mix of online and face-to-face (Zoom) seems to work exceptionally well. As we often work on different schedules and have to juggle work and other classes, the instantaneousness of online messaging is effective and provides a way of constant communication throughout the project. When more important meetings are required to discuss risk, strategy or reviewing large pieces of work, the group uses Zoom calls to provide a flowing conversation where body language and delivery of words tend to have more importance. Limiting our communication channels to two (Messenger group and Zoom), provides a simple communication structure with limited complexity.

Managing communications is another crucial part of communication and reporting as it aims to systematically share project information to the right people, at the right time and in a useful format. In doing so, the group should leverage technology and performance reporting whilst making use of the three key communication methods: 1) interactive communication 2) push communication and 3) pull communication. As the team in this project is rather small, primarily methods 1 and 2 will be used, through channels previously discussed. As mentioned above, performance reporting is also an important part of managing communications and quality as it allows stakeholders to provide important updates to the group in a structured and logical manner. These might take the form of status reports, progress reports and

forecasts, all of which are designed to keep stakeholders informed about how resources are being used to achieve project objectives. As a way to combine planning and managing, we have built a communication and reporting plan to stay on top of communication (included below).

Communication	Method	Frequency	Goal	Owner	Audience
Project status report	Message	Weekly	Review project status and discuss potential issues or delays	Abrar	Project team
Team standup	Meeting	Weekly	Discuss what each team member did yesterday, what they'll do today, and any blockers	Tom	Project team
Project review	Meeting	At milestones	Present project deliverables, gather feedback, and discuss next steps	Jian	Project team + project supervisors
Post-mortem meeting	Meeting	At end of project	Assess what worked and what did not work and discuss actionable takeaways	Abrar	Project team
Task progress updates	Message	Daily	Share daily progress made on project tasks and other key points to discuss	Tom	Project team

Finally, controlling communication will aim to ensure optimal flow of information throughout the life of the project, through systematic assessment and reflection of how well communications are managed. In general, communication will be reflected upon as a group to assess the following points: Can we develop better communication skills as individuals? Are our meetings and catch ups being run effectively? Can we make better use of technology and other tools to improve communication? In answering these questions on a weekly or fortnightly basis, the group will be able to effectively control and manage communications to ensure success throughout the project.

Review and audit mechanisms

Review and audit mechanisms to be used in this project are designed to determine the status of work performed on a project to ensure it complies with the statement of work, including the scope and timing of the project. This will allow us to better understand where we are compared to where we should be in the project and let us know if things are okay and performing as planned. The project auditing process will have five key parts: 1) plan the audit 2) conduct the audit 3) summarise and discuss the audit as group 4) determine actions required based on audit 5) schedule follow up. This will allow crucial risks and issues to be surfaced early so they can be acted upon and rectified.

During the project review and audits, the key focus of the group will be to analyse imperative internal factors to minimise risks and ensure project success. These factors will include timing / keeping up with project schedule, quality of work, whether more information / training required, version control and documentation. These represent the fundamental factors on which a solid foundation for the project can be built and will be reviewed systematically.

Schedule and resource management

Schedule

Schedule is a sequence of activities over time. We used WBS in a form of table to list and divide all activities and work packages and a Gantt chart to show how each task is done over time as shown in *Appendix 3*. Our schedule has three phases. The first phase is where we focus on analysing the data and creating a model for predicting a heart disease. As shown in the figure (ganttchart), the first phase will take around two to three weeks. The reason behind a quick and short milestone is that most subtasks in the first phase can be done simultaneously. The second stage is where we start to build our website and UI and implement our analysis to it. Unlike the first phase, we decided to put more time on UI. This is due to some members not fully knowing the area of the website and UI. Lastly the third phase is where we do the testing and do a review of our project. Testing phase can be done whenever the UI has started to be made. By testing regularly, we decrease the complexity of the issues over time. Furthermore, the second and third phases could start together since we use an iterative life cycle, we will do a regular code review and project review throughout the 12 weeks.

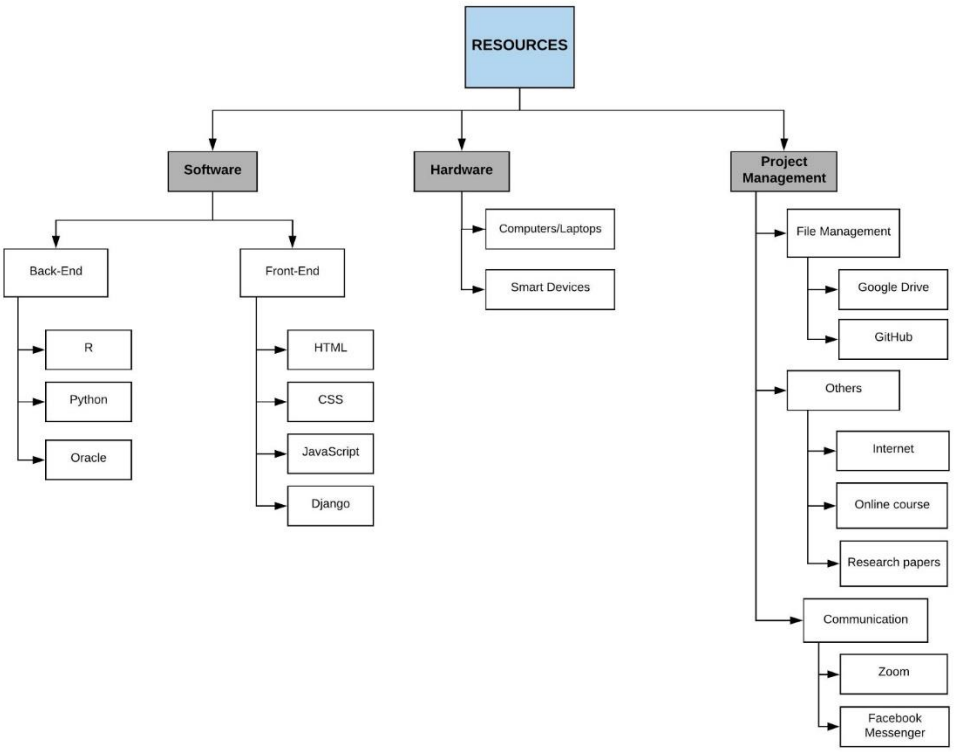
Resource requirements

Resource requirements are the supplies needed for a project goal to be achieved. Our resources are divided into three categories: software, hardware, and project management resources. As depicted in the figure below (resource breakdown structure), the hardware that we will mostly need to do in our phase one (Figure WBS + Gantt chart) are our own laptops or computers or any other electronics that supports writing codes and run analysis using R and python. This leads to our software that is needed in phase one which are R, python and we might use oracle for our data warehouse. These two languages are known for its capability to pre-process data and perform well in analysing and modelling.

Currently the second phase would be the most challenging work on our project since some of us lack knowledge in website making and user interface areas. The resources for us would differ than the first phase. On top of the computers and laptops, we might need smartphones or tablets. This is because we will create a website that is user friendly and easy to understand and interact through all platforms. As for our software, we are using HTML, CSS, JavaScript, and Django to create our interactive website. Project management resources will most likely be vastly different, we will extremely rely on the internet to find courses and similar projects on creating an interactive webpage. Furthermore, we barely know about security issues therefore, studying them through online courses will benefit us.

Regarding our third phase which is reviewing and keeping track of the project, for file management we will use google drive to share our files and reports, and GitHub to share our codes and programs throughout the project lifetime. As in human resources, all members are required to contribute equally to the project. Communication resources, at the moment, are through Facebook messenger and online meetings via Zoom. However, this is subject to change due to COVID-19. These are all the resources we might need; however, as the project

progresses, we might encounter more means to complete our project. Note a more in-depth requirement traceability matrix is provided in *Appendix 2.*



External design

Our goal in external design, is to create a web-based system that allows users easy access using the internet, with a beautiful and intuitive interface. Therefore, our audience will be anyone who can access the internet, with our software being for people who want to predict heart disease without having to go to the hospital and without costing them anything. As Minaei-Bidgoli, Kashy, Kortemeyer and Punch (2003) stated, a web-based system collects data in vast quantities in real-time since it uses the internet. This is the reason we decided on using a web-based system, as a way to gather more data allowing our model to better identify patterns and hidden information, and as a result better performance and accuracy.

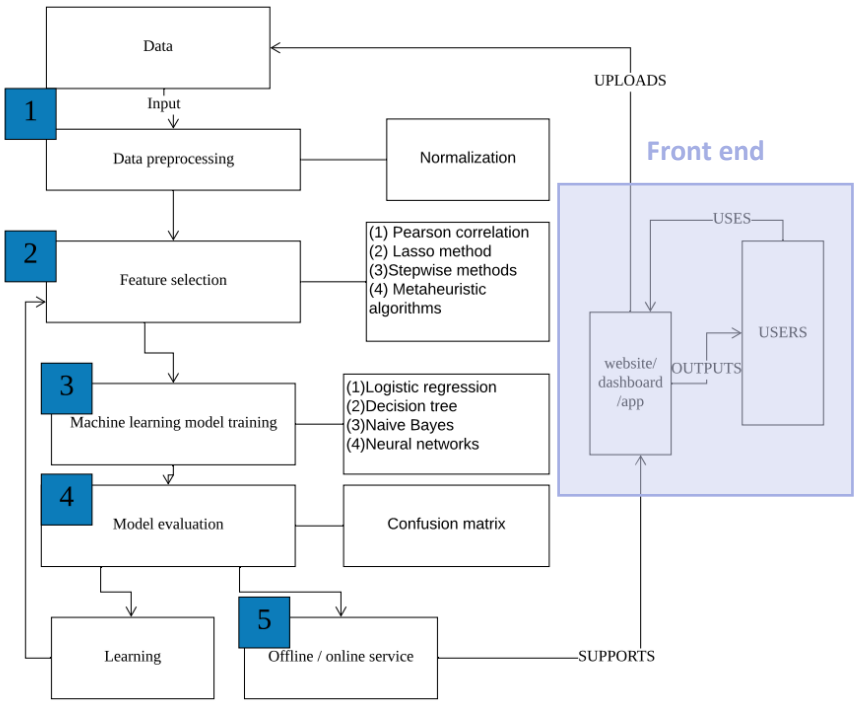
Our website will have three main pages: a page where information about heart disease will be provided, a page where users can check their heart and a page where resources, references and other things will be stated. In addition, we will have a hidden page that is accessible after the user submitted their data and to see their result. The main feature of our main page is we will provide facts about heart disease and provide some visuals to improve the attractiveness and interactivity aspect. We considered making the visuals to be interactive such as graphs that can be filtered through time and places. The third page does not have any special feature and it contains references, resources, and a reporting system if the user had difficulties.

The second page is where most functionalities underlie. The first feature that the user will see is the two options for inputting their data. They could input their data manually or by

uploading a file. However, our upload mechanism will only allow tabular format files such as csv and xlsx due to minimising error, security issues and maximising the performance of our software. In the background, the website will be connected to the data warehouse that we created. The idea is after a user has done inputting their data, the data will be stored in the data warehouse for further analysis using our machine learning model. The other special feature that our website gives is two output systems. The first option is for users who wanted to see the result by itself without diving into details. The second option is to see the detailed report. This latter option is aimed for doctors for analysing how our models work. These options will redirect users to the hidden page where all reporting and results will be displayed depending what the user selected for the output. Note UI designs have been included in *Appendix 5.*

Methodology

To complete this study, with the aim of building an accurate and reliable predictive product for coronary artery disease (CAD), a five-phase data analytic methodology has been used. The *first phase* - data pre-processing - will include two steps 1) data cleaning 2) data censoring. The *second phase* will aim to use the processed data from phase one to understand important features in the data and which are most important in classifying patients as CAD positive. This will be done through various feature selection methods which include Pearson correlation, Lasso & stepwise methods and metaheuristic algorithms. The *third phase* of classification will use the selected features from phase two to start building and training classification models.



The main methods used will include decision trees, neural networks, support vector machines which will be benchmarked against simpler, more traditional methods such as logistic regression and naive Bayes classifiers. Once the models have been built and trained, *phase four* will aim to evaluate each model and their predictions on various testing samples, and to build a weighted voting system between models to formalise a final prediction. This will finally be fed into *phase five* where a front end design will integrate the model to provide users with a complete and interactive experience. A summary of the above has been depicted in the diagram below.

Phase one - data pre-processing

The dataset used in this study has been collected by Z-Alizadehsani and contains samples of 303 patients, with each patient having their own set of 54 unique predictors. The predictors include important patient information such as demographic, symptoms, examinations and echo features, all of which allow for a prediction of the Boolean class variable "CAD".

Using the dataset described above, the first step of data pre-processing is to clean the data. In this study, we have used a derivative of the five-step data cleaning process described in Dag, Topuz, Oztekin, Bulur, Megahed's (2016) study. First, we will clean all erroneous and duplicated records using outlier detection algorithms within different R packages. Then we will eliminate any variables with no predictive power (i.e. patient number), as well as any invariant variables. Finally, missing records from the data will also be eliminated or imputed, depending on the magnitude. The second step in data processing will be to censor any sensitive data the patient might upload, mainly through the use of k-anonymity methods which will be implemented in the R package *sdcMicro*.

Phase two - Feature selection

Based on evidence in various studies, we have chosen four feature selection methods to better understand which variables in the data set are important in predicting CAD, and can be used as parameters to classification methods. The first is Pearson Correlation, a statistic that measures linear correlation between two or more variables and can be reproduced within algorithms to choose the best features. This method will be implemented using the various R packages and a program written to output most important features with relevant graphs and metrics.

Next is the Lasso method which is a form of penalised regression which uses an L1 norm as a regulariser and unlike Ridge regression, its norm regulariser drives parameters to zero, effectively deleting non-important variables. This will be implemented using the *sklearn* package in Python. Stepwise methods work in a similar way to Lasso, however the choice of predictive variables is carried out by an automatic procedure in which each step a variable is considered for addition to or subtraction from the set of explanatory variables based on some specific criterion. For this study we will use the Akaike (AIC) and Bayesian (BIC) information criterion to select features.

Finally, this study aims to leverage a novel approach to feature selection, through the use of metaheuristic algorithms and in particular the genetic algorithm. This algorithm is a

stochastic method for function optimisation based on the mechanics of natural genetics and biological evolution and will be implemented using the *caret* package in R.

Phase three - Classification

This phase will make the use of three fundamental data analytic models (decision trees, SVMs and neural networks), benchmarked against two conventional statistical methods (logistic regression and naive Bayes) to accurately classify each sample as CAD positive or negative.

Decision trees are one of the most understandable and easy ways to interpret data analytic methods, and are therefore used in several studies. In this study specifically, we will implement decision trees using the C&RT algorithm - a common tree building technique - due to its favourable performance compared to other tree algorithms previously analysed in the literature review and will implement this using the *rpart* package in R.

SVMs are supervised learning models with associated learning algorithms that analyse data given a set of training examples. As a way to perform both linear and non-linear classification, this study will use a method called the kernel trick, to implicitly map inputs into higher dimensional feature spaces. These will be implemented in R using the *libsvm* subset of the e1071 package.

The third fundamental classification method for this study uses various neural network methods (ANNs, LSTMs). These are described as computational systems made up of an input, hidden and output layer, which are composed of artificial neurons which built to combine input with their internal state and an optional threshold using an activation function, and as a result producing an output. These two methods will be implemented using the *neuralnet* and *rnn* packages in R.

The performance of the three methods described above will be benchmarked against two conventional methods (LR and NB), as a way to better understand the value of implementing more complex methods. Logistic regression will be implemented using the elastic net method which effectively combines Lasso & Ridge regression methods to produce a more accurate output. Naive Bayes classifiers will be implemented using the *e1071* package in R.

Phase four - Model evaluation

This phase aims to find an effective way to combine all the previous analysis and modelling into one final prediction which can be communicated to a user. This will leverage the use of various performance statistics (F1 score, AUC, sensitivity, specificity, precision and recall) to build a weighted scoring system between the different models used. In doing so, we will be able to leverage all of the predictive power of several methods with the aim of improving accuracy.

Phase five - Front-end implementation

The final phase aims to integrate the above predictive system into a complete and interactive user interface, to allow users to input their data to gain predictions. This will be

implemented through the use of the Django framework and R Markdown, and will include a function to add user data input back into the system to use as a way to continuously train the model moving forward.

Test planning

Test coverage

Testing of the product functionality and performance will be imperative for the success of this project. In doing so, test coverage will be divided into three key stages 1) data pre-processing 2) code implementation and 3) product delivery. Some information on these three stages has been described below.

- Data pre-processing stages
    - Test data integrity, consistency, accuracy in dataset
- Coding stages
    - Test the feasibility of different feature selection algorithms and classification algorithms using various edge cases to provide rigorous results
- Delivery Stages
    - Test whether the delivered product can run normally and can be interactive. Test ability to upload personal data to the database through the UI. Test whether a user can input information based on relevant keywords and get predictable results.

Test methods

Test priority (Low/Medium/High) proves to be an effective method, each test case should have a priority to ensure the smooth completion of the project. In the data pre-processing stage, we use a variety of test methods, for example, the evaluation of data quality / integrity to assess whether the data information is missing. Similarly, data consistency looks to assess whether the data follows a unified standard, and whether the data collection remains unified. The consistency of the format and data quality is mainly reflected in the specifications of the data records and whether the data is logical. Finally, data accuracy assesses whether the information recorded in the data is abnormal or incorrect, such as abnormally large or small data.

In the coding stage, we aim to use the Confusion Matrix to get accuracy, sensitivity and specificity (as well as other important metrics) to compare all models, also using AUC to help the evaluation. Unit testing will also be applied throughout the entire codebase, with a particular focus on functionality of the user interface and the integration of our models. A unit is the smallest testable part of any software. The test object is generally a function or class, in which various inputs and edge cases are tested to be certain of the output and functionality. The test basis for the final product will include the user acceptance criteria which has been previously described.

Sample Test Cases

Data to be collected

We will record the accuracy, sensitivity and specificity of different models, expected prediction results, actual prediction results and important features in each model. Test case ID, Test Designed By, Test Execution Date, Title/Name, Test Summary/Description, Status(Pass/Fail), Notes/Comments/Questions and Pre-conditions should also be collected.

Data storage and reporting

A summary of the key points in data storage and reporting, as well as an example have been included below:

- The error stage occurred should be recorded
- Validation error messages should be displayed properly in the correct position
- All error messages should be displayed in the same CSS style (For Example, using red colour)
- Test case format should be reported
- Each team member can see the current status of the test

Test case template						
Test design by: Jian Tan		Test design date:12/06/2020				
Test executed by :Jian Tan		Test execution date:				
Test case ID:	Test priority	Test Description	Expected Result	Actual result	Status	Comments:
T_1	High	Check if has null value	No any null value	Has two null value	Fail	Find it and delete
T_2						
T_3						

Conclusion

Throughout this project proposal document we have gathered all the required information from which to effectively design, build, test and maintain a working product. First, an update on recent research in the space was formalised through a literature review, where we better understood which key machine learning methods would prove most efficient for our use case. Next we extensively discussed how the project was going to be managed through a detailed project management plan, which provided insights into process model, scope and requirements. We then discussed the external design of the product and what was deemed as most important for users to have a complete and interactive experience, followed by the methodology discussing how everything from machine learning methods to front-end integration was going to be implemented. Finally, test planning was discussed with various ways to make sure we have adequate systems in place for the product and deliverables are rigorously tested.

In doing so, we are able to propose a complete project plan, with the aim to build a product which predicts CAD in patients, allowing for a cheap diagnosis with no operation or appointment required. This tool will allow many people around the world with limited resources to still get adequate medical help, and could be the defining point between a patient's life and death.

References

Al-Aidaroos, K. M., Bakar, A. A., & Othman, Z. (2012). Medical Data Classification with Naive Bayes Approach. *Information Technology Journal*, 1-6. doi:10.3923/itj.2012

Alizadehsani, R., Habibi, J., Hosseini, M. J., Mashayekhi, H., Boghrati, R., Ghandeharioun, A., ... & Sani, Z. A. (2013). A data mining approach for diagnosis of coronary artery disease. *Computer methods and programs in biomedicine*, *111*(1), 52-61.

Alizadehsani, R., Zangooei, M. H., Hosseini, M. J., Habibi, J., Khosravi, A., Roshanzamir, M., ... & Nahavandi, S. (2016). Coronary artery disease detection using computational intelligence methods. *Knowledge-Based Systems*, *109*, 187-197.

Aneja, S., & Lal, S. (2014). Effective Asthma Disease Prediction Using Naive Bayes – Neural Network fusion technique. *International Conference on Parallel*, 137-140. doi:10.1109/PDGC.2014.7030730

Babaoğlu, I., Fındık, O., & Bayrak, M. (2010). Effects of principle component analysis on assessment of coronary artery diseases using support vector machine. *Expert Systems with Applications*, *37*(3), 2182-2185.

Ghiasi, M. M., Zendehboudi, S., & Mohsenipour, A. A. (2020). Decision tree-based diagnosis of coronary artery disease: CART model. *Computer Methods and Programs in Biomedicine*, *192*, 105400.

Guo, C., Zhang, J., Liu, Y., Xie, Y., Han, Z., & Yu, J. (2020). Recursion Enhanced Random Forest With an Improved Linear Model (RERF-ILM) for Heart Disease Detection on the Internet of Medical Things Platform. *IEEE Access*, *8*, 59247-59256.

Hannan, S. A., Manza, R. R., & Ramteke, R. J. (2010). Generalized Regression Neural Network and Radial Basis Function for Heart Disease Diagnosis. *International Journal of Computer Applications, 7*(13), 7-13. doi: 10.5120/1325-1799

Jamain, A., & Hand, D. J. (2005). The Naive Bayes Mystery: A classification detective story. *Pattern Recognition Letters, 26*, 1752-1760. doi:10.1016/j.patrec.2005.02.001

Kahramanli, H., & Allahverdi, N. (2008). Design of a Hybrid System for the Diabetes and Heart Disease. *Expert Systems with Application, 35*, 82-89. doi:10.1016/j.eswa.2007.06.004

Minaei-Bidgoli, B., Kashy, D. A., Kortemeyer, G., & Punch, W. F. (2003). Predicting Student Performance: An application of data mining methods with an educational web-based system. *33rd Annual Frontiers in Education*, T2A-13. doi:10.1109/FIE.2003.1263284

Nikhar, S., & Karandikar, A. M. (2016). Prediction of Heart Disease Using Machine Learning Algorithms. *International Journal of Advanced Engineering, Management and Science, 2*(6), 617-621. Retrieved from https://www.neliti.com/publications/239484/prediction-of-heart-disease-using--learning-algorithms#cite

Raja, B. S., & Asghar, S. (2020). Beyond the Horizon: A Meticulous Analysis of Clinical Decision-Making Practices. *International Journal of Advanced Computer Science and Applications, 11*(2), 691-702. doi:10.14569/IJACSA.2020.0110287

Saritas, M. M., & Yasar, A. (2019). Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification. *International Journal of Intelligent Systems and Applications in Engineering, 7*(2), 88-91. doi:10.18201/ijisae.2019252786

Soni, J., Ansari, U., Sharma, D., & Soni, S. (2011). Predictive data mining for medical diagnosis: An overview of heart disease prediction. *International Journal of Computer Applications*, *17*(8), 43-48.

Tayefi, M., Tajfard, M., Saffar, S., Hanachi, P., Amirabadizadeh, A. R., Esmaeily, H., ... & Ghayour-Mobarhan, M. (2017). hs-CRP is strongly associated with coronary heart disease (CHD): A data mining approach using decision tree algorithm. *Computer methods and programs in biomedicine*, *141*, 105-109.

Venkatalakshmi, B., & Shivsankar, M. V. (2014). Heart Disease Diagnosis Using Predictive Data Mining. *International Journal of Innovative Research in Science, Engineering and Technology, 3*(3), 1873-1877. Retrieved from https://www.semanticscholar.org/paper/Heart-Disease-Diagnosis-Using-iveData-mining-B.Venkatalakshmi-kar/141e9fbeba0a737073d9d0b7bdeab83e666d1778#citing-papers

Zangooei, M. H., & Jalili, S. (2012). PSSP with dynamic weighted kernel fusion based on SVM-PHGS. *Knowledge-Based Systems*, *27*, 424-442.

Appendices

Appendix 1: Risk register

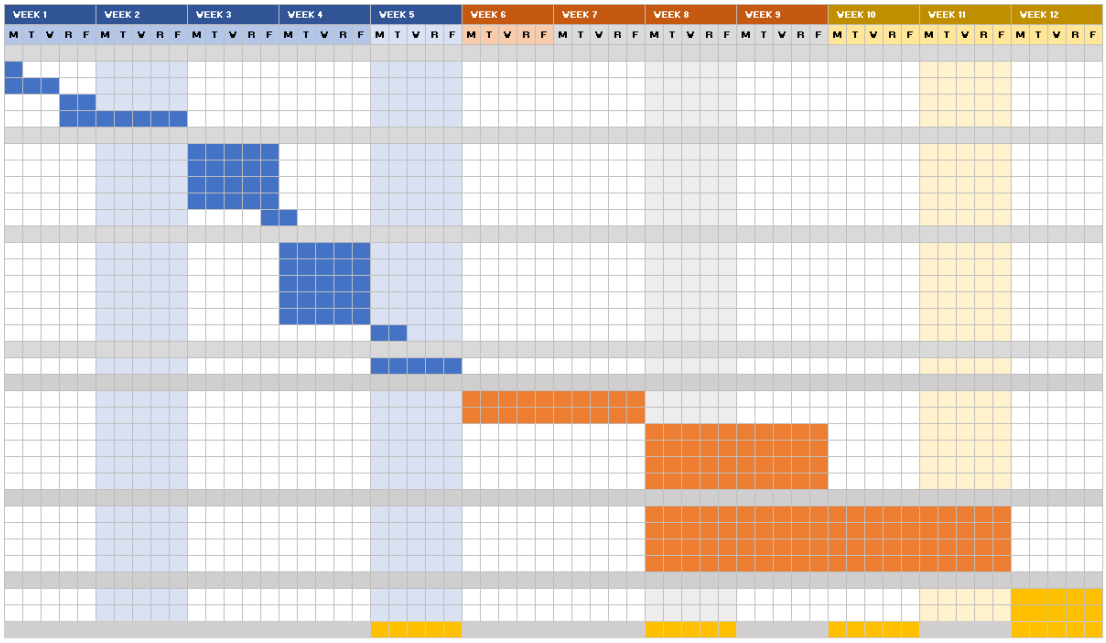
Detecting heart disease - Risk register												
Prepared by:	Tom Orlando, Jian Tan, Abrar Hamzah											
Date:	13-Jun-20											
Number	**Rank**	**Risk**	**Description**	**Category**	**Root cause**	**Triggers**	**Potential responses**	**Risk owner**	**Probability**	**Impact**	**Risk Score**	
001	4	Poor input and output quality	This looks at risks related to the quality of project inputs and outputs. In this case, the project inputs which include work from the our team and participation from external users whilst the output can include both the website and model	Product risk	Lack of skills / resources	Underperforming team members / low participation	Discuss how we can improve or simplify models and UI	Jian Tan	25%	8	2	
002	3	Falling behind on project schedule	Looks at uncertainty of forward looking estimates and assumptions relating to the project schedule, and when the project might complete	Schedule risk	Innacurate estimates	Falling behind on tasks	Reducing scope and deliverables or allocating resources to promote speed	Abrar Hamzah	40%	6	2.4	
003	9	Low quality of hardware and resources	Potentially having low quality hardware which leads to strain on project processes and implementations. Some models need efficient hardware to run.	Hardware risk	Inadequate resources	Models not running properly	Make use of university computers or borrow someone elses when required	Jian Tan	10%	6	0.6	
004	1	Linking back-end and front-end	Team members have limited front-end / UI development experience. This could lead to issues when implementing models into final product	Software risk	Lack of skills	Faulty implementation of models in the front end and breaking the website	Prepare before starting the project next semester and reach out for help whenever required	Tom Orlando	50%	6	3	
005	8	Low communication within group	Inadequate communication can lead to issues in functionalities and management of project. Can often be quite detrimental if not rectified	Communication risk	Low levels of communication	Trouble efficiently managing project and timelines	Set clear rules initially, and make sure to consistently reach out to team mates and use teachers and tutors if problem persists	Abrar Hamzah	20%	7	1.4	
006	5	Accidentally expanding the scope of project	Often occurs when wanting to fix problems or adding in functionalities, the scope can get too large and the project resources are stretched thin	Scope risk	Original scope didn't take into consideration some issues	Occurs when trying to fix issues by expanding the scope	Maintain a strict focus on scope and attempt to find other ways to solve issues	Tom Orlando	30%	6	1.8	
007	2	Attempting methods which are too complicated	Some of the methods planned can be quite complex to initialise and could lead to high complexity within the project	Skill risk	Attempted to use methods that are too complex	Falling behind on deadlines and building faulty models	Make a decision quickly as to whether or not a complex method is worth keeping	Jian Tan	40%	7	2.8	
008	7	Poor compatibility	Product is not compatible in different platforms.The technician did not consider the configuration of the computer so that some users could not log in normally	Product risk	Lack of skills	Faulty implementation or lack of testing	Compatibility testing on different devices, OS and web browsers, regularly	Abrar Hamzah	30%	5	1.5	
009	6	Security issues	Product is vulnerable	Product risk	Lack of skills	Occurs when user input values or upload a file	Security testing	Tom Orlando	40%	4	1.6	

Appendix 2: Requirements traceability matrix

REQUIREMENTS TRACEABILITY MATRIX						
Project Name:	Cardiovascular disease predictive modeling project					
Project Manager Name:	Jian Tan, Tom Orlando, Abrar Hamzah					
Project Description:	The aim of this project is to design and implement various data mining and predictive modelling methods, in an attempt to predict the occurrence of coronary artery disease (CAD) in patients, based on a particular set of parameters.					
ID	***Requirements (Functional or Non-Functional)***	***Assumption(s) and/or Customer Need(s)***	***Category***	***Source***	***Status***	
001	Interactive UI	Preferably support a website with user friendly and easy to understand	front-end requirement	Users	In Progress	
002	login to the application or website	We have different login interfaces with users	front-end requirement	Team member	In Progress	
003	Data input	Data can be entered correctly according to keywords	front-end requirement	Users	In Progress	
004	Algorithms status	Algorithms run properly, correctly, and efficiently	back-end requirement	Users	In Progress	
005	platforms compatibility	provide different platforms compatibility for users	hardware requirement	Users	In Progress	
006	Ensure data warehouse is properly connected to & imported in the IDEs	In case of problems, timely possible errors and suggestions for modification	back-end requirement	Users	In Progress	
007	Pre-process data	Will not cause abnormal errors due to null values, missing values and duplicate values	back-end requirement	Team member	In Progress	

Appendix 3: Work breakdown structure

WBS NUMBER	TASK TITLE	Dependency	TASK OWNER	DURATION	% of TASK COMPLETE
1	Data Pre-processing	Start		2 Week	
1.1	Importing all libraries & dataset		Jian		0%
1.2	Dealing with missing & duplicate values	FS [1.1]	Jian		0%
1.3	Normalisation	FS [1.1]	Jian		0%
1.4	Create datawarehouse	FS [1.1]		1 Week	0%
2	Feature Selection	Start & Finish after task [1]		1 Week	
2.1	Pearson Correlation	FS [1]	Abrar		100%
2.2	LASSO method	FS [1]	Tom		0%
2.3	Stepwise method	FS [1]	Abrar		0%
2.4	Metaheuristic algorithms	FS [1]	Tom		0%
2.5	Code review	FS [2]	Together		
3	Machine learning modeling	Start after [1], [2] & Finish after [4]		2 Week	
3.1	Logistic Regression	FS [2]	Tom		50%
3.2	Decision Tree	FS [2]	Jian		0%
3.3	SVM	FS [2]	Jian		0%
3.4	Naïve Bayes	FS [2]	Abrar		0%
3.5	Neural Network	FS [2]	Abrar		0%
3.6	Code review	SS [3]	Together		
4	Model Evaluation	Start when [3] starts		1 Week	
4.1	Compare performace of all algorithms	Dependent to ML models	Together		0%
5	Front End	Starts after Phase 1 is done		5 Week	
5.1.1	Website Making	Independent	Tom	1 Week	0%
5.1.2	Improve website appearance	SS [5.1.1]	Abrar	1 Week	0%
5.2	Adding UI functionality	FS [5.1]	Together	1 Week	0%
5.3*	UI security	FS [5.2]	Jian	2 Week	0%
5.4*	Data input security	FS [5.2]	Abrar	2 Week	0%
5.5*	UI performance optimisation	FS [5.2]	Tom	2 Week	0%
6	Testing & Feedback	Able to start when task [5.2] has started		1 Week	
6.1*	Data input testing	SS [5.4]	Abrar		0%
6.2*	UI performance testing	SS [5.5]	Tom		0%
6.3*	UI interactivity testing	SS [5.2]	Jian		0%
6.4*	Output testing	SS [5.4]	Abrar		
7	Deployment	Start when [1] to [6] are completed		1 Week	0%
7.1	Monitor performance	FS [7.7]	Together		0%
7.2	Monitor output	FS [7.7]	Together		0%
8	Project Review	FS [8]	Together		

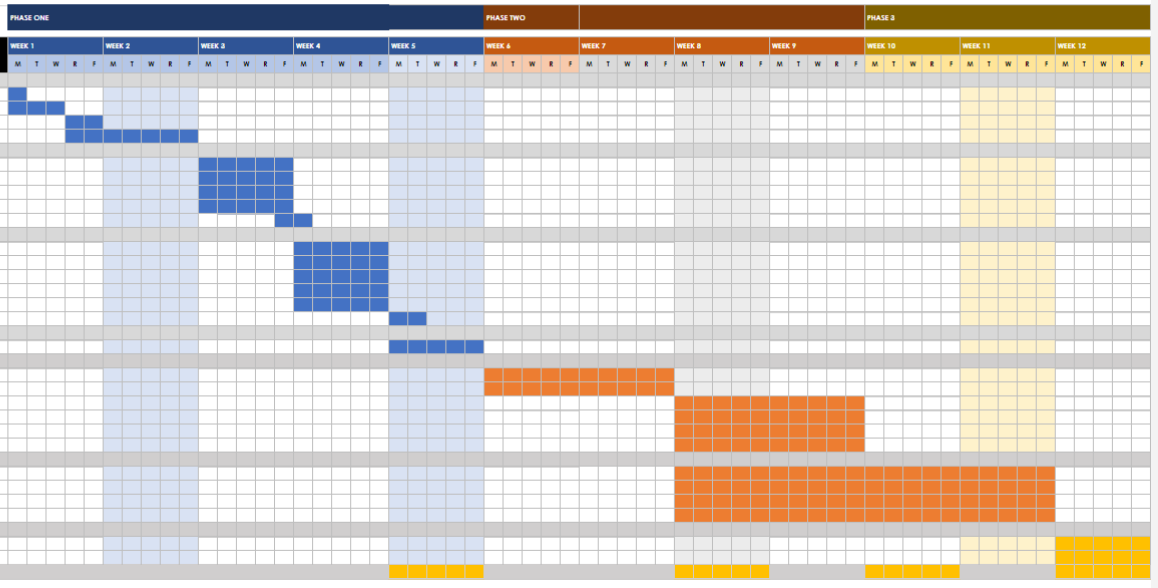


WORK BREAKDOWN STRUCTURE WITH GANTT CHART

PROJECT TITLE	Detecting heart disease
PROJECT MEMBERS	Abrar Hamzah, Jian Tan, Tom Orlando
DATE	8-Jun-20

WBS NUMBER	TASK TITLE	Dependency	TASK OWNER	DURATION	% of TASK COMPLETE
1	Data Pre-processing	Start		2 Week	
1.1	Importing all libraries & dataset		Jian		0%
1.2	Dealing with missing & duplicate values	FS [1.1]	Jian		0%
1.3	Normalisation	FS [1.1]	Jian		0%
1.4	Create datawarehouse	FS [1.1]		1 Week	0%
2	Feature Selection	Start & Finish after task [1]		1 Week	
2.1	Pearson Correlation	FS [1]	Abrar		100%
2.2	LASSO method	FS [1]	Tom		0%
2.3	Stepwise method	FS [1]	Abrar		0%
2.4	Metaheuristic algorithms	FS [1]	Tom		0%
2.5	Code review	FS [2]	Together		
3	Machine learning modeling	Start after [1], [2] & Finish after [4]		2 Week	
3.1	Logistic Regression	FS [2]	Tom		50%
3.2	Decision Tree	FS [2]	Jian		0%
3.3	SVM	FS [2]	Jian		0%
3.4	Naïve Bayes	FS [2]	Abrar		0%
3.5	Neural Network	FS [2]	Abrar		0%
3.6	Code review	SS [3]	Together		
4	Model Evaluation	Start when [3] starts		1 Week	
4.1	Compare performace of all algorithms	Dependent to ML models	Together		0%
5	Front End	Starts after Phase 1 is done		3 Week	
5.1.1	Website Making	Independent	Tom	1 Week	0%
5.1.2	Improve website appearance	SS [5.1.1]	Abrar	1 Week	0%
5.2	Adding UI functionality	FS [5.1]	Together	1 Week	0%
5.3*	UI security	FS [5.2]	Jian	2 Week	0%
5.4*	Data input security	FS [5.2]	Abrar	2 Week	0%
5.5*	UI performance optimisation	FS [5.2]	Tom	2 Week	0%
6	Testing & Feedback	Able to start when task [5.2] has started		1 Week	
6.1*	Data input testing	SS [5.4]	Abrar		0%
6.2*	UI performance testing	SS [5.5]	Tom		0%
6.3*	UI interactivity testing	SS [5.2]	Jian		0%
6.4*	Output testing	SS [5.4]	Abrar		
7	Deployment	Start when [1] to [6] are completed		1 Week	0%
7.1	Monitor performance	FS [7.7]	Together		0%
7.2	Monitor output	FS [7.7]	Together		0%
8	Project Review	FS [8]	Together		

*Most challenging task



Appendix 4: Team members' contribution

Literature review	
Jian	33%
Abrar	33%
Tom	33%
Project management	
Jian	33%
Abrar	33%
Tom	33%
External design	
Abrar	100%
Methodology	
Tom	100%
Test planning	
Jian	100%
Intro / conclusion	
Tom	100%

Appendix 5: UI design