

**DEPARTMENT OF INFORMATION TECHNOLOGY B.TECH IV YEAR I SEM SECTION-A A.Y:2022-23 MAJOR PROJECT ABSTRACT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TITLE OF MAJOR PROJECT: DETECTION OF CHRONIC HEART FAILURE(CHF) FROM HEART PULSATION THROUGH MACHINE LEARNING AND END-TO-END DEEP LEARNING** | | | | |
| **BAT**  **CH NO** | **ROLL NUMBER** | **STUDENT NAME** | **PHONE NUMBER** | **E-MAIL ID** |
| **04** | **19891A1221** | **CH.Y. SOWMYA SRI** | **7036763097** | **sowmyachowdarychil** [**kuri@gmail.com**](mailto:kuri@gmail.com) |
| **19891A1239** | **M. SAI KEERTHANA** | **7702233574** | **saikeerthanamedavar** [**apu@gmail.com**](mailto:apu@gmail.com) |
| **19891A1212** | **B. AKANKSHA** | **8341571223** | **akankshabhoomraog** [**ari@gmail.com**](mailto:ari@gmail.com) |
| **19891A1215** | **B. VAMSHI KRISHNA** | **9182308141** | **Vamshkrishna02002 @gmail.com** |
| **GUIDE NAME: G. SIRISHA** | | | | |
| **ABSTRACT:** Chronic heart failure (CHF) affects over 26 million of people worldwide, and its incidence is increasing by 2% annually. Despite the significant burden that CHF poses and despite the ubiquity of sensors in our lives, methods for automatically detecting CHF are surprisingly scarce,even in the research community. We present a method for CHF detection based on heart sounds. The method combines classic Machine-Learning (ML) and end-to-end Deep Learning (DL). The classic ML learns from expert features, and the DL learns from a spectro- temporal representationof the signal. The method was evaluated on recordings from 947 subjects from six publicly available datasets and one CHF dataset that was collected for this study, the proposed method achieved a score of 89.3, which is 9.1 higher than the challenge’s baseline method. The method’s aggregated accuracy is 92.9% (error of 7.1%); while the experimental results are not directly comparable, this error rateis relatively close to the percentage of recordings labeled as  “unknown” by experts (9.7%). Finally, we identified 15 expert features that are useful for building ML models to differentiate between CHF phases with an accuracy of 93.2%. The proposed method shows promising results both for the distinction of recordings between healthy subjects and patients and for the detection of different CHF phases. This may lead to the easier identification of new CHF patients and the  development of home-based CHF monitors for avoiding hospitalizations | | | | |

Project Guide Project Coordinator

**EN105HS/EN205HS: ENGLISH**

**UNIT –I**

**‘The Raman Effect’ from the prescribed textbook ‘English for Engineers’ published by Cambridge University Press.**

**Vocabulary Building**: The Concept of Word Formation --The Use of Prefixes and Suffixes. **Grammar:** Identifying Common Errors in Writing with Reference to Articles and Prepositions. **Reading:** Reading and Its Importance- Techniques for Effective Reading.

**Basic Writing Skills:** Sentence Structures -Use of Phrases and Clauses in Sentences- Importance of Proper Punctuation- Techniques for writing precisely – **Paragraph writing** – Types, Structures and Features of a Paragraph - Creating Coherence-Organizing Principles of Paragraphs in Documents.

**UNIT –II**

**‘Ancient Architecture in India’ from the prescribed textbook ‘English for Engineers’ published by Cambridge University Press. Vocabulary:** Synonyms and Antonyms.

**Grammar:** Identifying Common Errors in Writing with Reference to Noun-pronoun Agreement and Subject-verb Agreement.

**Reading:** Improving Comprehension Skills – Techniques for Good Comprehension

**Writing:** Format of a Formal Letter-**Writing Formal Letters** E.g.., Letter of Complaint, Letter of Requisition, Job Application with Resume.

**UNIT –III**

**‘Blue Jeans’ from the prescribed textbook ‘English for Engineers’ published by Cambridge University Press.**

**Vocabulary**: Acquaintance with Prefixes and Suffixes from Foreign Languages in English to form Derivatives-Words from Foreign Languages and their Use in English.

**Grammar:** Identifying Common Errors in Writing with Reference to Misplaced Modifiers and Tenses.

**Reading:** Sub-skills of Reading- Skimming and Scanning

**Writing**: Nature and Style of Sensible Writing- **Defining- Describing** Objects, Places and Events –

**Classifying**- Providing Examples or Evidence

**UNIT –IV**

**‘What Should You Be Eating’ from the prescribed textbook ‘English for Engineers’ published by Cambridge University Press.**

**Vocabulary**: Standard Abbreviations in English

**Grammar:** Redundancies and Clichés in Oral and Written Communication.

**Reading**: Comprehension**-** Intensive Reading and Extensive Reading

**Writing: Writing Practices-**-Writing Introduction and Conclusion - Essay Writing-Précis Writing.

**UNIT –V**

**‘How a Chinese Billionaire Built Her Fortune’ from the prescribed textbook ‘English for Engineers’ published by Cambridge University Press.**

**Vocabulary**: Technical Vocabulary and their usage

**Grammar:** Common Errors in English

**Reading:** Reading Comprehension-Exercises for Practice

**Writing: Technical Reports**- Introduction – Characteristics of a Report – Categories of Reports

**Formats- Structure of Reports (Manuscript Format) -Types of Reports - Writing aReport TEXT BOOK:**

1. Sudarshana, N.P. and Savitha, C. (2018). English for Engineers. Cambridge University Press.

**REFERENCE BOOKS:**

1. Swan, M. (2016). Practical English Usage. Oxford University Press.
2. Kumar, S and Lata, P. (2018). Communication Skills. Oxford University Press.
3. Wood, F.T. (2007). Remedial English Grammar. Macmillan.
4. Zinsser, William. (2001). On Writing Well. Harper Resource Book.
5. Hamp-Lyons, L. (2006). Study Writing. Cambridge University Press.

Exercises in Spoken English. Parts I –III. CIEFL, Hyderabad. Oxford University Press

**CS103ES/CS203ES: PROGRAMMING FOR PROBLEM SOLVING**

# UNIT - I: Introduction to Programming

Introduction to components of a computer system: disks, primary and secondary memory, processor, operating system, compilers, creating, compiling and executing a program etc., Number systems Introduction to Algorithms: steps to solve logical and numerical problems. Representation of Algorithm, Flowchart/Pseudo code with examples, Program design and structured programming Introduction to C Programming Language: variables (with data types and space requirements), Syntax and Logical Errors in compilation, object and executable code , Operators, expressions and precedence, Expression evaluation, Storage classes (auto, extern, static and register), type conversion, The main method and command line arguments

Bitwise operations: Bitwise AND, OR, XOR and NOT operators

Conditional Branching and Loops: Writing and evaluation of conditionals and consequent branching with if, if-else, switch-case, ternary operator, goto, Iteration with for, while, dowhile loops

I/O: Simple input and output with scanf and printf, formatted I/O, Introduction to stdin, stdout and stderr. Command line arguments

# UNIT - II: Arrays, Strings, Structures and Pointers:

Arrays: one- and two-dimensional arrays, creating, accessing and manipulating elements of arrays Strings: Introduction to strings, handling strings as array of characters, basic string functions available in C (strlen, strcat, strcpy, strstr etc.), arrays of strings Structures: Defining structures, initializing structures, unions, Array of structures

Pointers: Idea of pointers, Defining pointers, Pointers to Arrays and Structures, Use of Pointers in self- referential structures, usage of self referential structures in linked list (no implementation) Enumeration data type

# UNIT - III: Preprocessor and File handling in C:

Preprocessor: Commonly used Preprocessor commands like include, define, undef, if, ifdef,

ifndef Files: Text and Binary files, Creating and Reading and writing text and binary files,

Appending data to existing files, Writing and reading structures using binary files, Random access using fseek, ftell and rewind functions.

# UNIT - IV: Function and Dynamic Memory Allocation:

Functions: Designing structured programs, Declaring a function, Signature of a function, Parameters and return type of a function, passing parameters to functions, call by value, Passing arrays to functions, passing pointers to functions, idea of call by reference, Some C standard functions and libraries Recursion: Simple programs, such as Finding Factorial, Fibonacci series etc., Limitations of Recursive functions

Dynamic memory allocation: Allocating and freeing memory, Allocating memory for arrays of different data types

# UNIT - V: Introduction to Algorithms:

Algorithms for finding roots of a quadratic equations, finding minimum and maximum numbers of a given set, finding if a number is prime number, etc.

Basic searching in an array of elements (linear and binary search techniques), Basic algorithms to sort array of elements (Bubble, Insertion and Selection sort

algorithms), Basic concept of order of complexity through the example programs **TEXT BOOKS:**

1. Byron Gottfried, Schaum’s Outline of Programming with C, McGraw-Hill
2. B.A. Forouzan and R.F. Gilberg C Programming and Data Structures, Cengage Learning, (3rd Edition)

**REFERENCE BOOKS:**

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice 2. Hall of India

3. R.G. Dromey, How to solve it by Computer, Pearson (16th Impression) 4. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.

5. Herbert Schildt, C: The Complete Reference, Mc Graw Hill, 4th Edition

**CS404PC: DATABASE MANAGEMENT SYSTEMS**

# UNIT - I

**Database System Applications:** A Historical Perspective, File Systems versus a DBMS**,** the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS **Introduction to Database Design**: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model

# UNIT - II

**Introduction to the Relational Model**: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical data base design, introduction to views, destroying/altering tables and views.

Relational Algebra**,** Tuple relational Calculus, Domain relational calculus.

# UNIT - III

**SQL: QUERIES, CONSTRAINTS, TRIGGERS:** form of basic SQL query, UNION, INTERSECT,

and

EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active data bases.

**Schema Refinement**: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multi-valued dependencies, FOURTH normal form, FIFTH normal form.

# UNIT - IV

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions.

# UNIT - V

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree base Indexing, Comparison of File Organizations, Indexes and Performance Tuning, Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

**TEXT BOOKS:**

*1.* Database Management Systems, Raghurama Krishnan, Johannes Gehrke, *Tata Mc Graw Hill*

3rd Edition 2. Database System Concepts, Silberschatz, Korth, *Mc Graw hill*, V edition.

**REFERENCE BOOKS:**

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, *Pearson Education*
3. Introduction to Database Systems, C. J. Date, *Pearson Education*
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, *SPD*.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL,Shah, *PHI*. Fundamentals of Database Management Systems, M. L. Gillenson, *Wiley Student* Edition

**CS502PC: SOFTWARE ENGINEERING**

# UNIT - I

**Introduction to Software Engineering:** The evolving role of software, changing nature of software, software myths.

**A Generic view of process:** Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models.

**Process models:** The waterfall model, incremental process models, evolutionary process models, the unified process.

# UNIT - II

**Software Requirements:** Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

**Requirements engineering process:** Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

|  |
| --- |
| **System models:** Context models, behavioral models, data models, object models, structured |
| methods. |

# UNIT - III

**Design Engineering:** Design process and design quality, design concepts, the design model. **Creating an architectural design:** software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

# UNIT - IV

**Testing Strategies:** A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

**Product metrics:** Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

# UNIT - V

**Metrics for Process and Products:** Software measurement, metrics for software quality. **Risk management:** Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.

**Quality Management:** Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

**TEXT BOOKS:**

1. Software Engineering, A practitioner’s Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education.
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

**REFERENCE BOOKS:**

1. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
2. Software Engineering principles and practice- Waman S Jawadekar, The Mc Graw-Hill Companies.
3. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

**CS504PC: WEB TECHNOLOGIES**

# UNIT- I

**Introduction to PHP:** Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

**File Handling in PHP:** File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

# UNIT- II

**HTML Common tags**- List, Tables, images, forms, Frames; Cascading Style sheets;

**XML:** Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemes, Document Object Model, XHTML Parsing XML Data **–** DOM and SAX Parsers in java.

# UNIT - III

**Introduction to Servlets:** Common Gateway Interface (CGt), Life cycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

# UNIT - IV

**Introduction to JSP:** The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.

# UNIT - V

**Client-side Scripting:** Introduction to Javascript, Javascript language – declaring variables, scope of variables, functions. event handlers (onclick, onsubmit etc.), Document Object Model, Form validation.

**TEXT BOOKS:**

1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP — Steven Holzner, Tata McGraw-Hill

**REFERENCE BOOKS**

1. Web Programming, building internet applications, Chris Bates 2″ edition, Wiley Dreamtech
2. Java Server Pages —Hans Bergsten, SPD O’Reilly,
3. Java Script, D.Flanagan
4. Beginning Web Programming-Jon Duckett WROX.

**CS513PE: DATA ANALYTICS (Professional Elective - I)**

# UNIT - I

Data Management: Design Data Architecture and manage the data for analysis, understand various sources of Data like Sensors/Signals/GPS etc. Data Management, Data Quality(noise, outliers, missing values, duplicate data) and Data Processing & Processing.

# UNIT - II

Data Analytics: Introduction to Analytics, Introduction to Tools and Environment, Application of Modeling in Business, Databases & Types of Data and variables, Data Modeling Techniques, Missing Imputations etc. Need for Business Modeling.

# UNIT - III

Regression – Concepts, Blue property assumptions, Least Square Estimation, Variable Rationalization, and Model Building etc.

Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

# UNIT - IV

Object Segmentation: Regression Vs Segmentation – Supervised and Unsupervised Learning, Tree Building – Regression, Classification, Overfitting, Pruning and Complexity, Multiple Decision Trees etc. Time Series Methods: Arima, Measures of Forecast Accuracy,

STL approach, Extract features from generated model as Height, Average Energy etc and Analyze for prediction

# UNIT - V

Data Visualization: Pixel-Oriented Visualization Techniques, Geometric Projection Visualization Techniques, Icon-Based Visualization Techniques, Hierarchical Visualization Techniques, Visualizing Complex Data and Relations. **TEXT BOOKS:**

1. Student’s Handbook for Associate Analytics – II, III.
2. Data Mining Concepts and Techniques, Han, Kamber, 3rd Edition, Morgan Kaufmann Publishers.

**REFERENCE BOOKS:**

1. Introduction to Data Mining, Tan, Steinbach and Kumar, Addision Wisley, 2006.
2. Data Mining Analysis and Concepts, M. Zaki and W. Meira
3. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand Rajaraman Milliway Labs Jeffrey D Ullman Stanford Univ.

**CS615PE: SOFTWARE TESTING METHODOLOGIES (Professional Elective - III)**

# UNIT - I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs

Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

# UNIT - II

Transaction Flow Testing: transaction flows, transaction flow testing techniques. Dataflow testing: Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing. Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

# UNIT - III

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

# UNIT - IV

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

# UNIT - V

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like JMeter or Win-runner).

**TEXT BOOKS:**

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition. 2. Software Testing Tools – Dr. K. V. K. K. Prasad, Dreamtech.

**REFERENCE BOOKS:**

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.