A Mini Project Report

On

# ONLINE VOTING SYSTEM USING PHP AND MYSQL

*Submitted to JNTU HYDERABAD*

*In Partial Fulfillment of the requirements for the Award of Degree of*

**BACHELOR OF TECHNOLOGY IN**

**COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE)**

Submitted By

**E. AKHIL- 218R1A6721**

**M. SAITEJA -218R1A6743**

**B. ROHAN- 218R1A6713**

**V. TARUN- 218R1A6764**

Under the Esteemed guidance of

# Mrs. K DURGA

Assistant Professor, Department of CSE(DS)



# Department of Computer Science & Engineering (Data Science)

**CMR ENGINEERING COLLEGE UGC AUTONOMOUS**

(Approved by AICTE, NEW DELHI, Affiliated to JNTU, Hyderabad) Kandlakoya,

Medchal Road, R.R. Dist. Hyderabad-501 401.

**2024-2025**

CMR ENGINEERING COLLEGE

**UGC AUTONOMOUS**

*(Accredited by NBA,Approved by AICTE NEW DELHI, Affiliated to JNTU, Hyderabad) Kandlakoya, Medchal Road, Hyderabad-501 401*

**Department of Computer Science & Engineering(Data Science)**



**CERTIFICATE**

This is to certify that the project entitled **“ONLINE VOTING SYSTEM USING PHP AND MYSQL”**

is a Bonafide work carried out by

## E.AKHIL- 218R1A6721 M.SAITEJA -218R1A6743 B.ROHAN- 218R1A6713 V.TARUN- 218R1A6764

in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY**

### in COMPUTER SCIENCE AND ENGINEERING (DATA SCIENCE) from

CMR Engineering College, affiliated to JNTU, Hyderabad, under our guidance and supervision.

The results presented in this project have been verified and are found to be satisfactory. The results embodied in this project have not been submitted to any other university for the award of any other degree or diploma.

### Internal Guide Mini Project Head of the Department Mrs. k Durga coordinator Dr. M. Laxmaiah

Assistant Professor **Mrs. k Durga**

CSE(DS), CMREC Assistant Professor professor and HOD

CSE(DS), CMREC CSE(DS), CMREC

**DECLARATION**

This is to certify that the work reported in the present project entitled **“ONLINE VOTING SYSTEM**

**USING PHP AND MYSQL”** is a record of bonafide work done by us in the Department of Computer Science and Engineering (Data Science), CMR Engineering College, JNTU Hyderabad. The reports are based on the project work done entirely by us and not copied from any other source. We submit our project for further development by any interested students who share similar interests to improve the project in the future.

The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of our knowledge and belief.

**E. AKHIL- 218R1A6721**

**M. SAITEJA -218R1A6743**

**B. ROHAN- 218R1A6713**

**V. TARUN- 218R1A6764**

**ACKNOWLEDGMENT**

We are extremely grateful to **Dr. A. Srinivasula Reddy**, Principal and **Dr. M. Laxmaiah**, HOD,

**Department of CSE (Data Science), CMR Engineering College** for their constant support**.**

We am extremely thankful to **Mrs. K Durga,** Assistant Professor, Internal Guide, Department of CSE(DS), for his/ her constant guidance, encouragement and moral support throughout the project.

We will be failing in duty if I do not acknowledge with grateful thanks to the authors of the references and other literatures referred in this Project.

We thank **Mrs. K Durga** Mini Project Coordinator for his constant support in carrying out the project activities and reviews.

We express my thanks to all staff members and friends for all the help and co-ordination extended in bringing out this project successfully in time.

Finally, I am very much thankful to my parents who guided me for every step.

**. E. AKHIL- 218R1A6721**

**M. SAITEJA -218R1A6743**

**B. ROHAN- 218R1A6713**

**V. TARUN- 218R1A6764**

## CONTENTS

**TOPIC PAGE NO ABSTRACT**

1. INTRODUCTION 1-2
   1. Problem statement
2. LITERATURE SURVEY 3-5
   1. Motivation
3. EXISTING SYSTEM
   1. Disasvantages of existing system **6-7**
4. PROPOSED SYSTEM 8-11
   1. Advantages of proposed system
5. UML DIAGRAMS 12-16
   1. [Class Diagram](#_TOC_250003)
   2. [Use Case Diagram](#_TOC_250002)
   3. [Sequence Diagram](#_TOC_250001)
   4. [Activity Diagram](#_TOC_250000)
6. PHP and MYSQL 17-23
   1. Categories of php and mysql
   2. Challenges of php and mysql
   3. Applications of php and mysql
   4. Advantages of php and mysql
   5. Disadvantages of php and mysql
7. **SOFTWARE ENVIRONMENT** **24-42**
   1. What is python and Advantages of Python
   2. Disadvantages of python
   3. Modules used in project
   4. Installation of python
8. **SYSTEM REQUIREMENTS** **43**
   1. Software Requirements
   2. Hardware Requirements
9. **FUNCTIONAL REQUIREMENTS** **44-50**
   1. Output Design and Definition
   2. Input Design, Stages, Types, Media
   3. User Interface Design
10. **SOURCE CODE** **51-52**
11. **RESULTS AND DISCUSSION** **53**
12. **CONCLUSION AND REFERENCES** **54-55**

# ONLINE VOTING SYSTEM USING PHP AND MYSQL

**ABSTRACT**

The aim of this mini project is to develop an efficient and secure online voting system using PHP and MySQL. With the increasing importance of electronic voting systems, there is a pressing need for a reliable platform that ensures transparency, security, and ease of use. The proposed system incorporates features such as user authentication, candidate registration, ballot generation, secure vote casting, and result tabulation. The system utilizes PHP for server-side scripting and MySQL for database management, ensuring robustness and scalability. Security measures including encryption techniques, user authentication, and data validation mechanisms are implemented to safeguard the integrity of the voting process. The system also provides an intuitive user interface for voters to cast their votes securely from any internet-enabled device. Overall, the proposed online voting system offers a convenient and trustworthy solution for conducting elections in various domains while ensuring the integrity and confidentiality of the voting process**.**

creation, and secure vote casting, with results being automatically tabulated and displayed. Security is a major focus, incorporating mechanisms such as password encryption, data validation, and session management to prevent unauthorized access and ensure the integrity of each vote. The system addresses common issues in manual voting, such as long wait times, human errors in vote counting, and limited accessibility due to geographic Key features of the system include user authentication, candidate registration, ballot constraints.The online voting system improves voter participation by making the process accessible and easy to use, while eliminating manual errors associated with traditional paper ballots. It also reduces the time required for vote counting, offering near-instantaneous results. As a highly scalable solution, it can be adapted to different types of elections, from student councils to larger organizational or government elections.

# CHAPTER 1 INTRODUCTION

With the growing importance of digital technologies, there has been a significant shift toward the digitization of various sectors, including elections. Traditional voting systems, which rely on manual paper ballots, have been long used but are often fraught with inefficiencies, errors, and security concerns. Manual processes can be time-consuming, prone to human error, and susceptible to fraud or manipulation. In recent years, there has been a heightened demand for more reliable, efficient, and transparent systems that not only simplify the voting process but also ensure that each vote is accurately recorded and counted.

This online voting system is designed to allow voters to cast their ballots from any location with internet access, removing the geographical barriers that often prevent people from voting. The system leverages PHP for server-side programming, managing the logic behind user interactions, while MySQL serves as the back-end database for storing and managing election data, such as user credentials, candidates, and votes. Together, PHP and MySQL offer a robust and dynamic framework that ensures the application can handle large volumes of data while maintaining high levels of security and performance.

One of the core features of this system is user authentication, which is essential in ensuring that only eligible voters are able to access the platform and cast their votes. The system incorporates multiple security measures, including password encryption, data validation, and session management to prevent unauthorized access and fraudulent activities. Voters are required to register with valid identification, after which they can securely log in using their credentials. Once authenticated, voters can view a list of registered candidates, cast their votes, and receive confirmation that their vote has been successfully recorded.

The voting process itself is designed to be intuitive and user-friendly, providing a seamless experience for users with varying levels of technical proficiency. The system ensures that each voter can only cast their vote once, and all votes are securely stored in the MySQL database to maintain the integrity of the election.

**PROBLEM STATEMENT :**

Traditional voting systems, which primarily rely on manual processes such as paper ballots and in- person voting, are often inefficient, time-consuming, and prone to several critical issues. These problems include long wait times at polling stations, human errors in vote tallying, and vulnerabilities to fraud or manipulation. Additionally, the geographical limitation of having to visit a specific voting location can significantly reduce voter turnout, especially for individuals living in remote areas or those with mobility restrictions.

The **Online Voting System using PHP and MySQL** is designed to address these issues by providing a secure, scalable, and user-friendly platform for conducting elections over the internet. By allowing voters to cast their ballots from any internet-enabled device, this system eliminates the need for physical polling stations, reduces human errors in vote counting, and provides real- time results as soon as the voting period ends.

This project seeks to solve the following key problems:

1. **Accessibility Issues:** The geographical restrictions of traditional voting systems prevent a significant number of eligible voters from participating in elections, reducing overall voter turnout.
2. **Security Concerns:** Manual voting systems are vulnerable to security breaches, including ballot tampering, vote duplication, and unauthorized access to sensitive voter information.
3. **Efficiency and Accuracy:** The manual tallying of votes is labor-intensive and prone to human errors, resulting in delayed and potentially inaccurate election results.
4. **Time-Consuming Voting Process:** Long queues at polling stations, especially during peak voting hours, cause delays and frustration among voters, discouraging them from casting their ballots.

The **Online Voting System using PHP and MySQL** provides a solution that is scalable for different types of elections, enhances voter accessibility, ensures security, and delivers accurate, real-time results, thus addressing the critical issues inherent in traditional voting methods.

## CHAPTER 2 LITERATURE SURVEY

1. Online Voting System for India Based on AADHAR ID (2013) Himanshu Aggarwal and G. N.

Pandey proposed an online voting system specifically designed for India, utilizing the AADHAR ID for voter identification. This model aimed to address issues of voter impersonation by linking the voter’s identity to the national biometric database. The system ensured high levels of security but also raised concerns about privacy and the potential misuse of sensitive voter information. This study emphasized the importance of integrating national identification systems into the voting process to ensure voter authenticity.

1. Web-Based Voting System Using Fingerprint Design and Implementation (2019) In their study, Firas I. Hazzaa et al. developed a web-based voting system using fingerprint authentication to enhance security. The system leveraged web technologies to provide a practical solution to voting. The integration of biometric features ensured that the system could effectively authenticate users, preventing fraud or multiple votes by the same individual. While biometric systems offer high security, they can sometimes be challenging to implement on a large scale due to the requirement for specialized hardware.
2. An Efficient Online Voting System (2021) Anand A. and Divya P. developed an online voting system to facilitate secure elections for users, candidates, and election commission officers. Their work focused on improving voting percentage by allowing voters to participate in elections remotely. By incorporating user-friendly interfaces and real-time results, this system aimed to boost voter engagement. The system was praised for its scalability and ease of use, particularly in regions with high internet penetration. However, challenges related to security and vote tampering still persisted, as the system relied on traditional web technologies.
3. Secure Authentication for Online Voting System Using Biometrics and Steganography (2021)

Smita B. Khaimar and P. Sanyasi Naidu proposed an innovative online voting system that combined biometric authentication with steganography for enhanced security. Their system aimed to prevent unauthorized access and protect sensitive voter data. By embedding voter information within multimedia files through steganography, the system added an additional layer of security.While effective, the system’s complexity and the need for specialized tools limited its wide-scale adoption. The study highlighted the trade-off between security and system complexity in the design of online voting systems.

1. Blockchain-Based Voting Systems Recent advancements in blockchain technology have been explored for secure online voting. Blockchain offers a decentralized and transparent platform where each vote can be recorded immutably, ensuring that election results cannot be tampered with. Studies by Zyskind et al. (2018) and Yavuz et al. (2019) emphasized the benefits of blockchain-based voting systems, such as enhanced transparency, traceability, and security. However, the implementation of blockchain in voting systems also comes with challenges, such as scalability issues, complex infrastructure requirements, and the lack of public trust in emerging technologies.
2. Improving Security in Web-Based Voting Using PHP and MySQL (2018) This study by

**C. P. Nguyen et al.** explored ways to improve the security of web-based voting systems by enhancing both the **PHP code structure** and **MySQL database encryption techniques**. The research focused on addressing the common vulnerabilities of web-based systems, such as SQL injection, cross-site scripting (XSS), and man-in-the-middle attacks (MITM). The system incorporated security best practices, including **prepared statements** in PHP to prevent SQL injection, **HTTPS protocol** for secure communication, and **role-based access control** to restrict unauthorized users. The authors concluded that while PHP and MySQL offer a flexible framework for developing voting systems, proper implementation of security measures is crucial to prevent cyber-attacks and ensure the system's reliability.

## MOTIVATION:

The **Online Voting System using PHP and MySQL** is motivated by the desire to modernize the voting process and address the shortcomings of traditional methods. The increasing availability of internet access across the globe provides an opportunity to improve voter participation by allowing individuals to vote from the comfort of their homes or any location with an internet connection. This system eliminates geographical barriers and provides greater convenience to voters, especially for those who face challenges in reaching polling stations. In modern democracies, the voting process is fundamental to ensuring that the voices of citizens are heard. However, traditional voting systems, which rely on paper ballots and physical polling stations, are often plagued by inefficiencies, logistical challenges, and potential vulnerabilities to fraud. These systems can lead to long wait times, human errors in vote counting, and inaccessibility for voters living in remote areas or those with disabilities. Additionally, the costs associated with organizing physical polling stations and hiring personnel make the process resource-intensive. The need for a secure, accessible, and efficient voting process is more crucial than ever as societies increasingly move towards digital transformation. A key motivator behind developing an online voting system is the need to ensure **voting integrity and transparency**. Elections are prone to security risks such as ballot tampering, vote duplication, and unauthorized access to sensitive information. A well- designed online voting system, built using PHP and MySQL, offers the potential to mitigate these risks through the implementation of **user authentication mechanisms**, **encryption protocols**, and **secure database management**. By integrating these security features, the system can protect the confidentiality and integrity of votes while preventing unauthorized users from accessing the system.

## CHAPTER 3 EXISTING SYSTEM

The election process in India is the cornerstone of its vibrant democracy, ensuring the participation of citizens in choosing their representatives. Governed by constitutional provisions and election statutes, this structured process is pivotal in maintaining the integrity and efficiency of elections. In this comprehensive guide, we delve into the intricacies of the Indian election process, covering its various stages, schedules, and significant aspects.

### Election Process in India:

1. Announcement of Election Schedule by the ECI: The Election Commission of India (ECI) declares the schedule of elections through a formal announcement, setting the timeline for crucial events in the electoral process.
2. Issue of Notification by the ECI: Following the announcement, the ECI issues notifications calling upon the electorate to elect members of the legislative bodies.
3. Filing of Nominations by the Candidates: Candidates file their nominations in their respective constituencies within the stipulated timeframe after the issuance of notifications.
4. Oath or Affirmation of Candidates: Candidates make an oath or affirmation before authorized officers, signaling their commitment to contest in the elections.
5. Election Campaign: Political parties and candidates engage in election campaigns to garner support and persuade voters through various means, adhering to the Model Code of Conduct.
6. Allocation of Symbols: Candidates receive symbols, either reserved for national/state parties or allocated from a list of free symbols, for identification on the ballot.
7. Polling Days and Voting Procedure: Voting takes place over multiple days across different constituencies, conducted using Electronic Voting Machines (EVMs) to ensure secrecy and efficiency.
8. Supervising Elections and Media Coverage: The ECI deploys election observers to oversee the electoral process, while media coverage ensures transparency while respecting voter secrecy.
9. Counting of Votes and Constitution of House: Votes cast in EVMs are counted, and the candidate with the highest votes is declared the winner. The elected members form the legislative bodies, completing the electoral cycle... Read more at: [https://www.studyiq.com/articles/electoral-](https://www.studyiq.com/articles/electoral-system-in-india/) [system-in-india/](https://www.studyiq.com/articles/electoral-system-in-india/)

### Disadvantages of existing system

Vulnerability to Fraud: The existing voting system is vulnerable to fraud, which can undermine the integrity of elections. This includes manipulation of voting machines or software, tampering with ballot boxes or paper ballots, fake voting IDs or impersonation, and insider threats from election officials or staff.

Disenfranchisement: The potential for disenfranchisement exists due to voter ID laws, voter roll purges or errors, polling station closures or consolidation, and restrictions on early voting or absentee ballots.

Technical Complexity: Coercion or intimidation can occur through voter suppression tactics, threats or intimidation at polling stations, economic coercion, and social pressure from family or community members.

Long Wait Times: Long wait times are often due to inadequate polling station staffing or resources, high voter turnout or unexpected surges, technical issues with voting equipment, and inefficient voter check-in or verification processes.

Limited Candidate Information: Voters may lack access to comprehensive candidate information, including debates, public forums, media coverage, and online resources.

Systemic Exclusion Issues: Systemic exclusion issues persist, including barriers for citizens with disabilities, language barriers, and limited access to voting for homeless or transient individuals and rural or underserved communities.

Environmental Impact: The environmental impact of paper waste, energy consumption, and electronic waste from voting equipment is a concern.

Lack of Automation: The lack of automation leads to manual counting errors, time-consuming and labor-intensive voting processes, limited scalability, and inefficient voter registration or update processes.

Security Concerns: Security concerns include cybersecurity threats, physical security risks, insider threats, and potential voting machine hacking or manipulation.

Lack of Transparency: The lack of transparency in voting data, results, auditing, and verification processes erodes public confidence

## CHAPTER 4

**PROPOSED SYSTEM**

An **online voting system** is a web-based application that allows users to cast their votes through the internet. This type of system is widely used in various fields such as politics, education, and business, as it enables efficient and convenient voting processes. In this article, we will guide you through the steps of creating an online voting system in PHP, one of the most popular programming languages for web development.

To create a voting system in PHP, you need to have a good understanding of the language’s fundamentals and web development concepts. Here are the basic steps involved in creating a voting system:

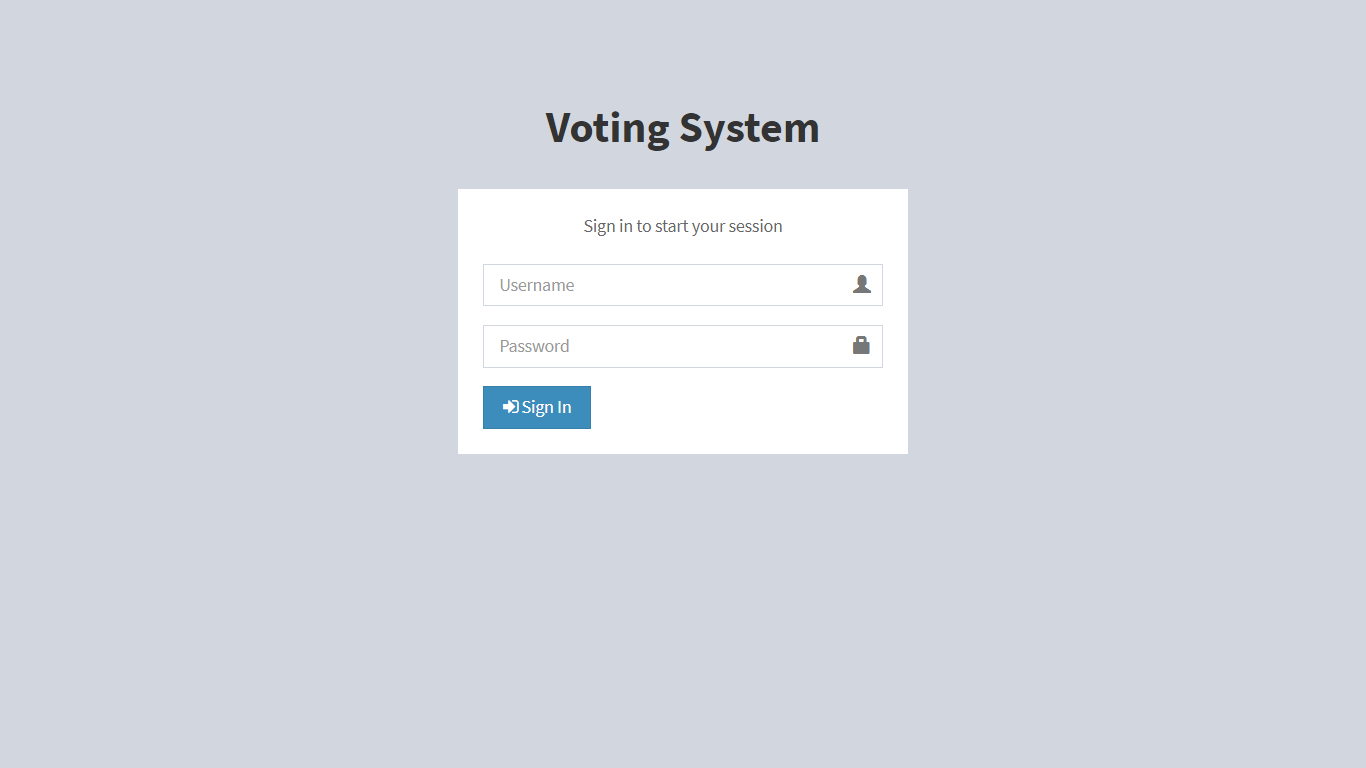
* **Define the requirements**. Define the purpose of your voting system, the type of voting method you want to use, and the security measures you need to implement.
  + The first step in creating an online voting system is designing the database schema. The database schema should include tables for storing information about voters, candidates, and the voting process. Here are some of the key tables that you should include in your database:
  + Users: This table should store information about voters, such as their name, email address, and login credentials.
  + Candidates: This table should store information about the candidates running for office, such as their name, party affiliation, and photo.
  + Votes: This table should store information about the votes cast by users, including the user ID, candidate ID, and the date and time of the vote.
* **Design the database**. Design the database schema and create the necessary tables to store user information, voting data, and other relevant information.
* **Develop the front-end**. Create a user-friendly interface for your voting system, including the ballot and voting options.
  + The next step is to develop the user interface for your online voting system. The user interface should be user-friendly and easy to navigate, allowing voters to easily find information about the candidates and cast their vote. Here are some of the key features that you should include in your user interface:
  + Candidate profiles: Each candidate should have a profile page that includes information about their background, platform, and stance on key issues.
  + Voting page: The voting page should allow users to cast their vote for their preferred candidate.
  + Results page: The results page should display the current vote count for each candidate and update in real-time as votes are cast.
* **Develop the back-end**. Develop the back-end of your voting system using PHP, including the logic for vote counting, result calculation, and data validation.

To implement polling in PHP, you need to develop a back-end script that collects the poll data from the users, stores it in a database, and calculates the poll results. Here are the basic steps involved in implementing polling in PHP:

* Create the poll form: Create a form that displays the poll question and the options to the user.
* Process the poll data: Process the poll data submitted by the user and store it in a database.
* Calculate the poll results: Calculate the poll results based on the data stored in the database and display the results to the user.

To create an authentication system in PHP, you need to develop a back-end script that authenticates the user’s credentials and provides access to the voting system. Here are the basic steps involved in creating an authentication system in PHP:

* + 1. **Design the database**. Design the database schema and create the necessary tables to store user information, login credentials, and other relevant information.
    2. **Develop the login form**. Develop a user-friendly login form that prompts the user to enter their login credentials.
    3. **Validate the login credentials**. Validate the user’s login credentials against the information stored in the database.
    4. **Set up user sessions**. Set up user sessions to maintain the user’s login state and provide access to the voting system.



### 4.2. Advantages of the proposed system

Enhanced Accessibility: Voters can participate from anywhere, at any time, making it easier for people with mobility issues, those living far from polling stations, or those with conflicting schedules to cast their votes .

Increased Efficiency: Automated vote counting eliminates manual errors, saving time and resources. The system can also handle large volumes of voters simultaneously, reducing wait times .

Improved Accuracy: Electronic voting minimizes errors caused by miscounted or misread paper ballots. The system ensures each voter only casts one vote, preventing fraud .

Real-time Results: The system can display real-time results, allowing for timely announcement of winners and reducing suspense .

Cost-Effectiveness: Online voting reduces costs associated with paper ballots, printing, transportation, and storage. It also minimizes the need for polling staff .

Environmental Benefits: Reduced paper usage helps minimize the ecological footprint of the voting process

Scalability: The system can easily accommodate growing voter numbers, making it ideal for large- scale elections .

Security: Implementing encryption, secure protocols, and password protection ensures the integrity of the voting process .

Accessibility: Allows voters to participate from anywhere with internet access, increasing voter turnout and engagement

Accuracy: Eliminates manual counting errors and reduces the risk of ballot tampering or fraud through encryption and secure authentication measures.

Convenience: Voters can cast their ballots at any time during the voting period, reducing long queues and wait times associated with traditional voting methods.

11

## CHAPTER 5 UML DAIGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software systems, as well as for business modeling and other nonsoftware systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML uses mostly graphical notations to express the design of software projects.

**GOALS:** The Primary goals in the design of the UML are as follows:

* Provide users with a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
* Provide extensibility and specialization mechanisms to extend the core concepts.
* Be independent of particular programming languages and development processes.
* Provide a formal basis for understanding the modeling language.
* Encourage the growth of the OO tools market.
* Support higher-level development concepts such as collaborations, frameworks, patterns and components.

### Class diagram

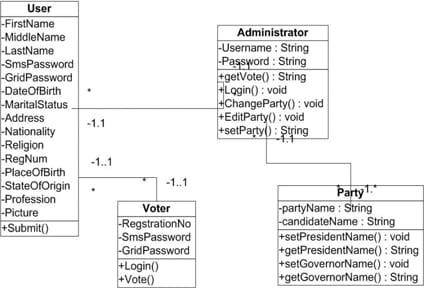
A class diagram is a graphical representation of the structure and relationships between classes in an object-oriented programming (OOP) system. It's a fundamental tool for software design,

development, and documentation.

12

Definition: A class diagram is a visual representation of:

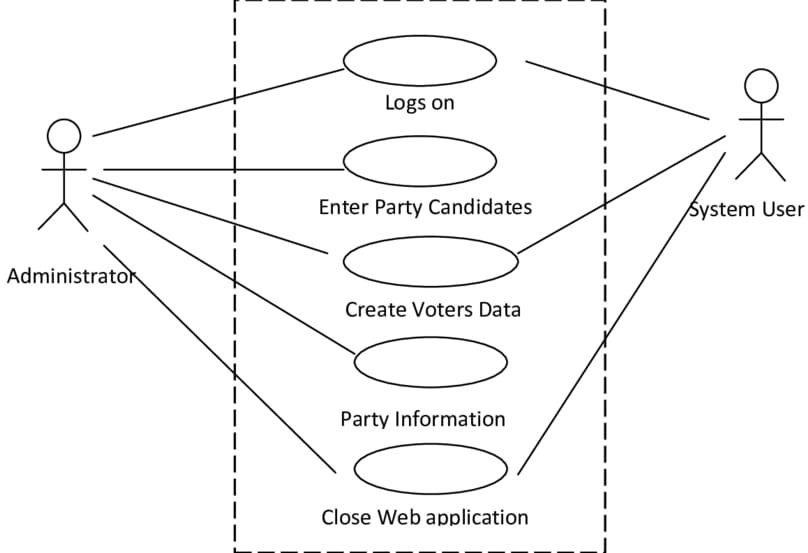
1. Classes: Representing entities, concepts, or objects in the system.
2. Attributes: Data members or properties of each class.
3. Methods: Functions or operations performed by each class.
4. Relationships: Connections between classes, such as inheritance, association, aggregation, and composition.



13

### Use case Diagram

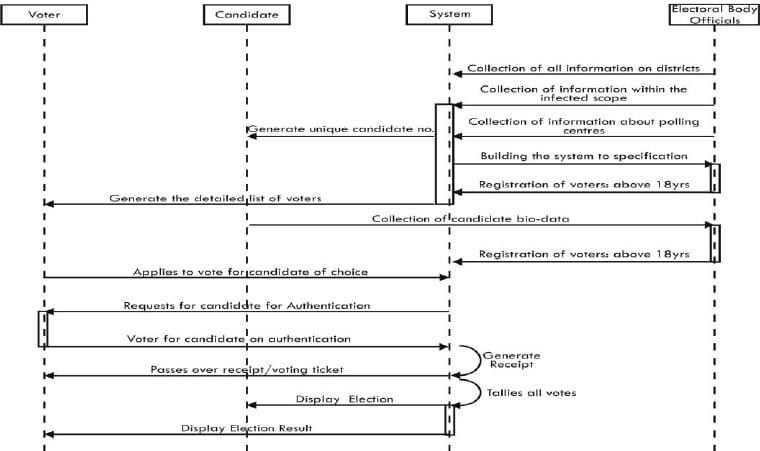
A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



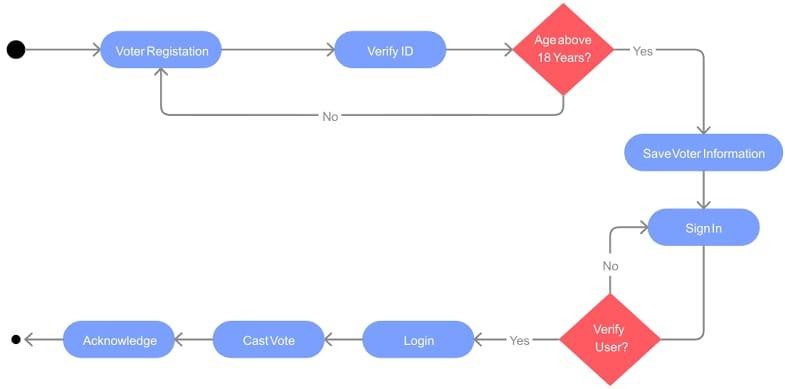
14

### Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows, as parallel vertical lines ("lifelines"), different processes or objects that live simultaneously, and as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.



### Activity diagram



# CHAPTER 6

**PHP AND MYSQL**

### What is PHP and MYSQL

PHP: Hypertext Preprocessor

PHP is a server-side scripting language, originally designed for web development, that allows developers to create dynamic web applications. It is an open-source, cross-platform language that can be used for web development, including web applications, content management systems, e-commerce platforms, social networking platforms, and more.

MySQL is an open-source Relational Database Management System (RDBMS) that allows users to store, organize, and manage data in a structured and scalable manner. It is a popular database management system used for web applications, content management systems, e- commerce platforms, social networking platforms, and more.

### Categories of PHP and MYSQL

PHP Categories :PHP is a versatile programming language that falls under several categories. Firstly, it is a server-side scripting language, allowing developers to create dynamic web applications. Additionally, PHP is widely used in web development, content management systems (CMS), e-commerce platforms, social networking platforms, blogging platforms, and web services (APIs). PHP can also be used for command-line scripting, desktop applications, and mobile app development.

PHP Applications: PHP powers some of the most popular web applications, including WordPress, Joomla, Drupal, Magento, Laravel, CodeIgniter, and Symfony. These applications demonstrate PHP's capabilities in building complex web systems, from blogging platforms to e-commerce websites. PHP's flexibility and customizability make it an ideal choice for developers.

MySQL Categories: MySQL is a relational database management system (RDBMS) that falls under several categories. It is primarily used for database design, data modeling, data warehousing, business intelligence, data analytics, web database management, cloud

database management, mobile app database management, and enterprise database management. MySQL's scalability and reliability make it a popular choice for large-scale database applications.

MySQL Applications: MySQL powers some of the most popular web applications, including Facebook, Twitter, YouTube, Wikipedia, eBay, Amazon, LinkedIn, WordPress.com, Joomla.org, and Drupal.org. These applications demonstrate MySQL's capabilities in handling large volumes of data and supporting complex web systems.

PHP and MySQL Integration:

PHP and MySQL are often used together to create dynamic web applications. Their integration enables developers to build scalable and secure web systems. PHP frameworks like Laravel and CodeIgniter provide tools for working with MySQL databases. Content management systems like WordPress and Joomla rely on MySQL for data storage. E- commerce platforms like Magento and osCommerce use MySQL for product catalogs and order management. The combination of PHP and MySQL enables developers to build complex web applications that require robust database management.

### Challenges in PHP and MYSQL

PHP Challenges:

1. Security vulnerabilities: PHP's open-source nature makes it vulnerable to security threats.
2. Performance issues: PHP can be slow, especially with complex applications.
3. Compatibility problems: PHP versions and configurations can cause compatibility issues.
4. Error handling: PHP's error handling can be challenging.
5. Code maintenance: PHP code can become complex and difficult to maintain.

MySQL Challenges:

1. Scalability limitations: MySQL can struggle with very large datasets.
2. Performance issues: MySQL query optimization can be challenging.
3. Data consistency: Ensuring data consistency across multiple databases can be difficult.
4. Security vulnerabilities: MySQL is vulnerable to SQL injection attacks.
5. Backup and recovery: MySQL backup and recovery processes can be complex.

PHP and MySQL Integration Challenges:

1. Connection management: Managing connections between PHP and MySQL can be tricky.
2. Query optimization: Optimizing MySQL queries for PHP applications can be challenging.
3. Data typing: Ensuring data type consistency between PHP and MySQL can be difficult.
4. Error handling: Handling errors across PHP and MySQL can be complex.
5. Performance tuning: Tuning PHP and MySQL performance together requires expertise.

Common Issues:

1. SQL injection attacks
2. Cross-site scripting (XSS) attacks
3. Cross-site request forgery (CSRF) attacks
4. Session hijacking
5. Database corruption

### Applications of PHP and MYSQL

PHP Applications:

1. Websites
2. Web Apps
3. Blogs
4. E-commerce sites
5. Social Media
6. Forums
7. Online Games
8. Mobile Apps
9. Desktop Apps
10. Web Services MySQL Applications:
    1. Database Management
    2. Web Applications
    3. E-commerce sites
    4. Social Media
    5. Blogs
    6. Online Forums
    7. Data Analysis
    8. Business Intelligence
    9. Data Storage
    10. Web Portals

Industries using PHP and MySQL:

1. E-commerce
2. Education
3. Healthcare
4. Finance
5. Media
6. Technology
7. Government
8. Non-profit
9. Real Estate
10. Travel

### Advantages of PHP and MYSQL

1. Dynamic web application development
2. Fast data processing and retrieval
3. Secure data storage and management
4. Scalable and flexible solutions
5. Easy integration with other technologies
6. Cost-effective development
7. Rapid development and deployment
8. High performance and reliability
9. Extensive community support
10. Wide range of applications
11. Cross-platform compatibility
12. Secure and reliable
13. Scalable and flexible
14. Cost-effective development
15. Opportunities for freelance work

### Disadvantages of PHP and MYSQL

PHP and MySQL Combination Disadvantages:

1. Connection Management Challenges: Managing connections between PHP and MySQL can be tricky.
2. Query Optimization Difficulties: Optimizing MySQL queries for PHP applications can be challenging.
3. Data Typing Inconsistencies: Ensuring data type consistency between PHP and MySQL can be difficult.

Mitigating Disadvantages:

1. Follow secure coding practices.
2. Use prepared statements and parameterized queries.
3. Optimize database queries.
4. Implement error handling and logging.
5. Regularly update PHP and MySQL versions.

# CHAPTER 7 SOFTWARE ENVIRONMENT

### What is Python?

Below are some facts about Python.

* + - Python is currently the most widely used multi-purpose, high-level programming language.
    - Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.
    - Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.
    - Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

The biggest strength of Python is huge collection of standard libraries which can be used for the following –

* + - Machine Learning
    - GUI Applications (like Kivy, Tkinter, PyQt etc.)
    - Web frameworks like Django (used by YouTube, Instagram, Dropbox)
    - Image processing (like Opencv, Pillow)
    - Web scraping (like Scrapy, BeautifulSoup, Selenium)
    - Test frameworks
    - Multimedia

### Advantages of Python

Let’s see how Python dominates over other languages.

1. Extensive Libraries : Python downloads with an extensive library and it contains code for various purposes like regular expressions, documentation generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don’t have to write the complete code for that manually.
2. Extensible : As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.
3. Embeddable : Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add scripting capabilities to our code in the other language.
4. Improved Productivity : The language’s simplicity and extensive libraries render programmers more productive than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.
5. IOT Opportunities : Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet of Things. This is a way to connect the language with the real world.
6. Simple and Easy : When working with Java, you may have to create a class to print ‘Hello World’. But in Python, just a print statement will do. It is also quite easy to learn, understand, and code. This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.
7. Readable : Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and indentation is mandatory.
8. Object-Oriented : This language supports both the procedural and object-oriented programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the encapsulation of data and functions into one.
9. Free and Open-Source : Python is freely available. But not only can you download Python for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.
10. Portable : When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn’t the same with Python. Here, you need to code only once, and you can run it anywhere. This is called Write Once Run Anywhere (WORA). However, you need to be careful enough not to include any system- dependent features.
11. Interpreted : Lastly, we will say that it is an interpreted language. Since statements are executed one by one, debugging is easier than in compiled languages.

Any doubts till now in the advantages of Python? Mention in the comment section.

### Advantages of Python Over Other Languages

1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don’t have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

1. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

The 2019 Github annual survey showed us that Python has overtaken Java in the most popular programming language category.

1. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and machine learning, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

### Disadvantages of Python

So far, we’ve seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let’s now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in slow execution. This, however, isn’t a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

1. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the client-side. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called Carbonnelle.

The reason it is not so famous despite the existence of Brython is that it isn’t that secure.

1. Design Restrictions

As you know, Python is dynamically-typed. This means that you don’t need to declare the type of variable while writing the code. It uses duck-typing. But wait, what’s that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can raise run-time errors.

1. Underdeveloped Database Access Layers

Compared to more widely used technologies like JDBC (Java DataBase Connectivity) and ODBC (Open DataBase Connectivity), Python’s database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

1. Simple

No, we’re not kidding. Python’s simplicity can indeed be a problem. Take my example. I don’t do Java, I’m more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

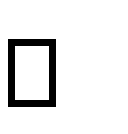
### History of Python

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde &Informatica). The greatest achievement of ABC was to influence the design of Python. Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners1, Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wiskunde en Informatica (CWI). I don't know how well people know ABC's influence on Python. I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it. "Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better33 properties, but without its problems. So I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or beginend blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

### Python Development Steps

Guido Van Rossum published the first version of Python code (version 0.9.0) at alt.sources in February 1991. This release included already exception handling, functions, and the core data types of lists, dict, str and others. It was also object oriented and had a module system. Python version

1.0 was released in January 1994. The major new features included in this release were the functional programming tools lambda, map, filter and reduce, which Guido Van Rossum never liked. Six and a half years later in October 2000, Python 2.0 was introduced. This release included

list comprehensions, a full garbage collector and it was supporting unicode. Python flourished for another 8 years in the versions 2.x before the next major release as Python 3.0 (also known as "Python 3000" and "Py3K") was released. Python 3 is not backwards compatible with

Python 2.x.

Print is now a function.

* + - Views and iterators instead of lists
    - The rules for ordering comparisons have been simplified. E.g., a heterogeneous list cannot be sorted, because all the elements of a list must be comparable to each other.
    - There is only one integer type left, i.e., int. long is int as well.
    - The division of two integers returns a float instead of an integer. "//" can be used to have the "old" behaviour.
    - Text Vs. Data Instead of Unicode Vs. 8-bit

### Purpose

We demonstrated that our approach enables successful segmentation of intra-retinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

### Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

* + - Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

### Modules Used in Project TensorFlow

TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache 2.0 open-source license on November 9, 2015.

### NumPy

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

* + - A powerful N-dimensional array object
    - Sophisticated (broadcasting) functions
    - Tools for integrating C/C++ and Fortran code
    - Useful linear algebra, Fourier transform, and random number capabilities.

31

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary datatypes can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

**Pandas:** Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

### Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object-oriented interface or via a set of functions familiar to MATLAB users.

### Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. Python features a dynamic type

32

system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

* + - Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
    - Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
    - Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

### Install Python Step-by-Step in Windows and Mac

Python a versatile programming language doesn’t come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.

The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Windows.

33

### How to Install Python on Windows and Mac

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

34

Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your System Requirements. Based on your system type i.e., operating system and based processor, you must download the python version. My system type is a Windows 64-bit operating system. So, the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheatsheet here. The steps on how to install Python on Windows 10, 8 and 7 are divided into 4 parts to help understand better.

### Download the Correct version into the system

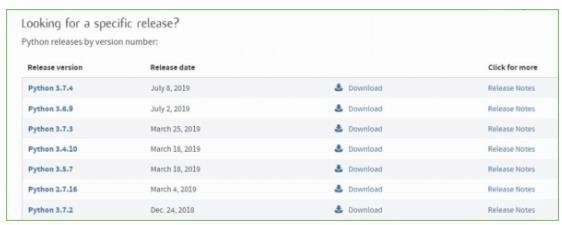
Step 1: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: https:/[/www](http://www.python.org/).[python.org](http://www.python.org/)



Now, check for the latest and the correct version for your operating system. Step 2: Click on the Download Tab.

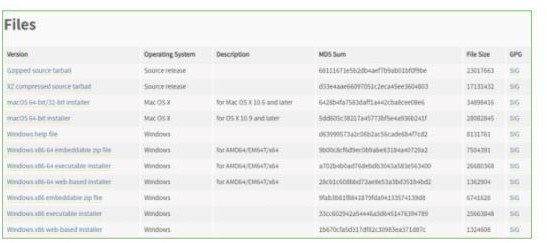


Step 3: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4



Step 4: Scroll down the page until you find the Files option.

Step 5: Here you see a different version of python along with the operating system.



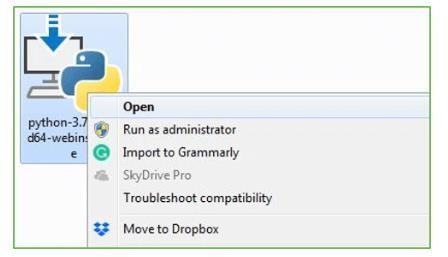
* To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.
* To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows
* x86-64 web-based installer.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e., Installation

Note: To know the changes or updates that are made in the version you can click on the Release Note Option.

### Installation of Python

Step 1: Go to Download and Open the downloaded python version to carry out the installation process.



Step 2: Before you click on Install Now, make sure to put a tick on Add Python 3.7 to PATH.



Step 3: Click on Install NOW After the installation is successful. Click on Close.

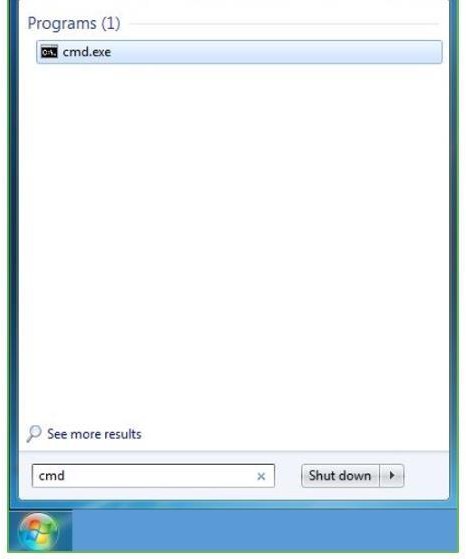


With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

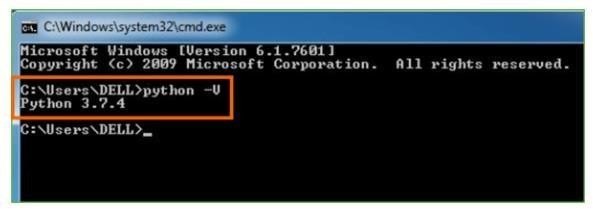
Note: The installation process might take a couple of minutes. Verify the Python Installation

Step 1: Click on Start

Step 2: In the Windows Run Command, type “cmd”.



Step 3: Open the Command prompt option. 38 Step 4: Let us test whether the python is correctly installed. Type python –V and press Enter.

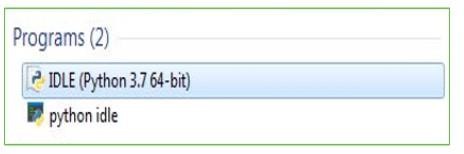


Step 5: You will get the answer as 3.7.

Note: If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

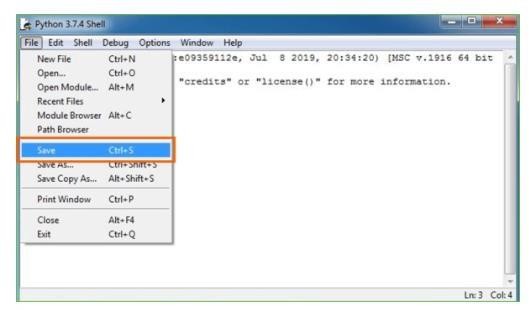
Check how the Python IDLE works Step 1: Click on Start

Step 2: In the Windows Run command, type “python idle”.



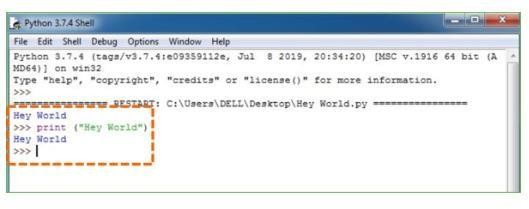
Step 3: Click on IDLE (Python 3.7 64-bit) and launch the program

Step 4: To go ahead with working in IDLE you must first save the file. Click on File > Click on Save.



Step 5: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

Step 6: Now for e.g., enter print (“Hey World”) and Press Enter.



You will see that the command given is launched. With this, we end our tutorial on how to install Python. You have learned how to download python for windows into your respective operating system.

Note: Unlike Java, Python does not need semicolons at the end of the statements otherwise it won’t work.

# CHAPTER 8 SYSTEM REQUIREMENTS

### Software Requirements

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation.

The appropriation of requirements and implementation constraints gives the general overview of the project in regard to what the areas of strength and deficit are and how to tackle them.

* + - Python IDLE 3.7 version (or)
    - Anaconda 3.7
    - Jupiter (or)
    - Google colab

### Hardware Requirements

Operating system : Windows, Linux Processor : minimum intel Processor: i5 and above

Ram: 8GB and above

Hard Disk: 500GB in local drive

43

# CHAPTER 9 FUNCTIONAL REQUIREMENTS

### Output Design

Voter Interface:

1. Login/Registration Form
2. Voter Dashboard
3. Voting Page
4. Candidate List
5. Voting Results Admin Interface:
   1. Login/Registration Form
   2. Admin Dashboard
   3. Candidate Management
   4. Voter Management
   5. Election Management
   6. Voting Results

Output Design Requirements:

Voting Page:

1. Display candidate names and photos
2. Display voting options (e.g., radio buttons, checkboxes)
3. Display voting instructions
4. Display "Cast Vote" button
5. Display "Cancel" button Voting Results:
   1. Display total votes cast
   2. Display candidate-wise vote count
   3. Display winner(s) of the election
   4. Display voting percentage
   5. Display interactive charts/graphs (e.g., bar chart, pie chart) Candidate List:
      1. Display candidate names
      2. Display candidate photos
      3. Display candidate profiles (e.g., bio, manifesto)
      4. Display candidate contact information Voter Dashboard:
         1. Display voter profile information
         2. Display voting history
         3. Display upcoming elections
         4. Display voting instructions Admin Dashboard:
            1. Display election schedule
            2. Display candidate list
            3. Display voter list
            4. Display voting results
            5. Display system notifications Reports:

Voting results report

Voter turnout report

Candidate performance report

Election summary report Security Requirements:

Secure login and authentication

Data encryption

Access control

Audit trail

Secure voting protocol

### 9.2. Input Design

Voter Registration Form

Input fields for name, email, password, and contact number. Option for uploading ID proof or any relevant documents. Login Form

Input fields for email and password. CAPTCHA for extra security.

Vote Casting Form

A list of candidates with radio buttons or dropdown to select the preferred candidate. A "Submit Vote" button to cast the vote.

Admin Election Creation Form

Input fields for election title, description, and date range. Candidate management (add candidate fields: name, party, etc.). Output Design

Voter Dashboard

Displays available elections, voting status (voted or not), and upcoming elections. Provides a link to cast the vote.

Admin Dashboard

Election management options, list of all voters, and a link to view results. Graphical display (charts) of votes counted for each candidate.

Voting Confirmation

A confirmation message or receipt after a successful vote submission. Error messages if voting fails or has already been done.

Election Results Display

Display the total number of votes each candidate received.

Bar charts or pie charts to represent voting distribution visually.

### 9.3. User Interface Design

The User Interface (UI) design of an online voting system built with PHP and MySQL should focus on being simple, intuitive, and secure for both voters and admins. Here is a breakdown of how to design the UI for such a system:

1. Login Page Components:

A clean login form with two input fields:

Email: User enters their email.

Password: User enters a secure password. Buttons:

"Login" button.

"Register" button for new users.

A link to "Forgot Password" for password recovery. Security: Add a CAPTCHA for bots prevention.

Design Style:

Minimalist, professional look.

Use labels for inputs and a visually distinct "Login" button.

Place the "Register" link or button below the login form, with subtle contrasting colors to avoid confusion.

1. Registration Page Components:

Input fields:

Full Name Email

Password (with password strength indicator) Confirm Password

Optionally, Date of Birth, and ID upload Security: CAPTCHA for security verification. Buttons: "Register" button at the end.

Terms and Conditions agreement checkbox. Design Style:

Ensure form fields are aligned vertically with appropriate spacing for ease of use. Color-coded form validation: Green for success, red for errors.

1. Voter Dashboard Components:

Welcome message (e.g., “Welcome, [Username]”).

Navigation Bar: Options like "Profile", "Elections", "Voting History", and "Logout". Current Elections Section:

Display elections that the user is eligible to vote in.

Each election listed with its name, date, and a "Vote Now" button. Voting History Section:

List of elections participated in with dates and results (if available). Option to view candidates from past elections.

Notifications/Announcements (optional).

Design Style:

Keep the design clean with clear sections.

Elections should be displayed as cards or panels, each with a short description, voting status (if voted already), and a CTA (Call to Action) button like “Vote Now”.

1. Voting Page Components:

Election Name and Details displayed at the top. Candidate list:

Each candidate with a name, party affiliation, and a brief description (or photo). Radio buttons or a single-selection dropdown for choosing the preferred candidate. Buttons: "Submit Vote" button.

50

# CHAPTER 10

## SOURCE CODE

candidate\_1 = input("Enter the 1st candidate name: ") candidate\_2 = input("Enter the 2nd candidate name: ") print(" ")

voters\_id = [101, 102, 103, 104, 105, 106, 107, 108, 109, 110]

cand\_1 = 0

cand\_2 = 0

no\_of\_voters = len(voters\_id) print("No. of voters:", no\_of\_voters) print(" ")

voted = []

while True:

if not voters\_id: print("Voting is over!") if cand\_1 > cand\_2:

print(f"{candidate\_1} won the election with {cand\_1} votes.") elif cand\_2 > cand\_1:

print(f"{candidate\_2} won the election with {cand\_2} votes.") else:

print("It's a tie!")

51

break else:

voter = int(input("Enter your voter ID: ")) if voter in voted:

print("You already voted!") else:

if voter in voters\_id:

print(f"1. {candidate\_1}\n2. {candidate\_2}") choice = int(input("Enter your choice: "))

if choice == 1: cand\_1 += 1

print(f"You voted for {candidate\_1}.") print(" ")

elif choice == 2: cand\_2 += 1

print(f"You voted for {candidate\_2}.") print(" ")

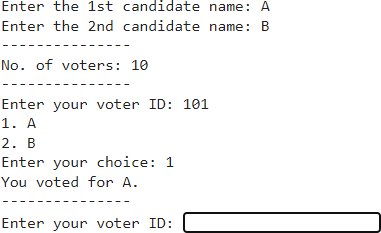
voters\_id.remove(voter) voted.append(voter)

else:

print("You are not allowed to vote.") print(" ")

# CHAPTER 11 RESULTS AND DISCUSSION

Based on the system requirements specification ONLINE VOTING INFORMATION SYSTEM has been designed and implemented



# CHAPTER 12 CONCLUSION

This Online Voting system will manage the Voter's information by which voter can login and use his voting rights. The system will incorporate all features of Voting system. It provides the tools for maintaining voter's vote to every party and it count total no. of votes of every party. There is a DATABASE which is maintained by the ELECTION COMMISION OF INDIA in which all the names of voter with complete information is stored.

In this user who is above 18 year's register his/her information on the database and when he/she want to vote he/she has to login by his id and password and can vote to any party only single time. Voting detail store in database and the result is displayed by calculation. By online voting system percentage of voting is increases. It decreases the cost and time of voting process. It is very easy to use and It is vary less time consuming. It is very easy to debug

## REFERENCES

1. Firas I. Hazzaa, Seifedine Kadry, Oussama Kassem Zein “Web-Based Voting System Using Fingerprint Design and Implementation” Vol. 2, Issue.4, Dec 2019
2. Himanshu Aggarwal and G.N. Pandey “Online Voting System for India Based on AADHAAR ID” 2013 Eleventh International Conference on ICT and Knowledge Engineering.
3. Anand A, and Divya P, “An efficient online voting system” in International Journal of Modern Engineering Research, Vol. 2(4), 2631–2634
4. Smita B. Khaimar, P. Sanyasi Naidu, Reena Kharat “Secure Authentication for Online Voting System”.
5. A. K. Singh, R. K. Singh, "Design and Implementation of Online Voting System Using PHP and MySQL", International Journal of Advanced Research in Computer Science, 2017.

https:/[/ww](http://www.ijarcs.info/index.php/Ijarcs/article/view/4424)w[.ijarcs.info/index.php/Ijarcs/article/view/4424](http://www.ijarcs.info/index.php/Ijarcs/article/view/4424)

1. S. S. Iyengar, N. R. Narayanan, "Online Voting System Using PHP and MySQL: A Secure Approach", International Journal of Computer Applications, 2018.

https://doi.org/10.5120/ijca2018917416

1. M. A. Alam, M. M. Rahman, "E-Voting System Using PHP and MySQL", Journal of Computer Science and Information Technology, 2019.
2. Luke Welling, Laura Thomson, "PHP and MySQL Web Development", Pearson Education, 2016.https:/[/www](http://www.amazon.com/PHP-MySQL-Web-Development-Luke-Welling/dp/0135189616).[amazon.com/PHP-MySQL-Web-Development-Luke-Welling/dp/0135189616](http://www.amazon.com/PHP-MySQL-Web-Development-Luke-Welling/dp/0135189616)
3. Tony Marston, "Building Online Voting Systems with PHP and MySQL", Packt Publishing, 2018. https:/[/ww](http://www.packtpub.com/product/building-online-voting-systems-with-php-and-mysql/978178728)w[.packtpub.com/product/building-online-voting-systems-with-php-and-mysql/978178728](http://www.packtpub.com/product/building-online-voting-systems-with-php-and-mysql/978178728)