RESUME BUILDER WEBSITE

BAKLIWAL FOUNDATION

College of Arts, Commerce & Science

A Project Report
Submitted in partial fulfillment of the
Requirements for the award of the Degree of

Bachelor of Computer Applications (BCA)
of
Kavikulaguru Kalidas Sanskrit University

Submitted by

ANURAG JAGATLAL GUPTA

ADITYA VIKAS NIKAM

B.K.SURAJ HARISH SINGH

Under the esteemed guidance of
Prof. Sneha Shashikant Lokhande

Kavikulaguru Kalidas Sanskrit University's

Bakliwal Foundation College of Arts, Commerce & Science

Vashi

BATCH: 2022-2025

Kavikulaguru Kalidas Sanskrit University's

Bakliwal Foundation College of Arts, Commerce & Science
Vashi

CERTIFICATE

This is to certify that the project entitled **Resume Builder Website** undertaken at the PCP Center: Bakliwal Foundation College of Arts, Commerce & science, Vashi, Navi Mumbai by **MR. BK SURAJ HARISH SINGH** holding **Seat No. (PRN:2022018100102204)** Studying **Bachelor of Computer Applications** Semester – VI has been satisfactorily completed as prescribed by the Kavikulaguru Kalidas Sanskrit University, during the year 2024 - 2025.

Project Incharge **Co-ordinator**

External Examiner

Internal Examiner **Principal**

ACKNOWLEDGEMENT

I would like to take this opportunity to express my deepest gratitude to **Professor Sneha Lokhande** for her invaluable guidance, support, and encouragement throughout the development of my project,. Her extensive knowledge and expertise in web development played a crucial role in shaping the direction and success of this project.

Professor Lokhande consistently provided thoughtful feedback, constructive criticism, and insightful suggestions that helped me refine my ideas and enhance the application's functionality at every stage—from initial concept to final implementation. Her ability to simplify complex concepts allowed me to gain a deeper understanding of the development process and strengthen the technical skills essential for my future professional growth.

I am sincerely grateful for her patience and support during the challenging phases of this project, which kept me motivated and focused on achieving my goals. Her mentorship has been instrumental in the successful completion of the Groceryz application and has inspired me to further pursue my interest in web development.

I would also like to extend my heartfelt thanks to**, Prof. Ankit Srivastava** for his helpful suggestions and support during the course of this project. His input was valuable in overcoming technical challenges and improving key components of the application.

I extend my sincere thanks to **Dr. Sharadkumar Shah, H.O.D Prof. Sneha Shashikant Lokhande, Prof. Shaikh Mohammed Umar, Prof. Divya Patil, Prof. Kalyani Kulkarni.**

Lastly, I would like to thank my friends for their constant encouragement, collaboration, and moral support. Their presence and insights contributed significantly to maintaining a positive mindset and staying on track throughout the project journey.

Once again, my sincere thanks to everyone who contributed in any way to the successful completion of this project.

Thanking you

DECLARATION

I hereby declare that the project entitled, “” done at **Bakliwal Foundation College of Arts, Commerce and Science, Vashi, Navi Mumbai**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree Of **BACHELOR OF COMPUTER APPLICATION** to be submitted as final semester project as part of our curriculum.

BK Suraj Harish Singh

Table of Contents

Sr No.	Title	Page No.
1	**Abstract**	1
1.1	Introduction Of Project	
2	**Literature Survey**	6
3	**Acquisition of Knowledge**	8
4	**Domain Knowledge**	12
4.1	Introduction	
4.2	Advantages of Domain Knowledge	
5	**System Study**	15
5.2	Benefits of proposed system	
6	**Problem Definition & Scope of Project**	18
6.1	Objective	
7	**Requirement Analysis**	21
7.1	Analysis	
7.2	Feasibility Study	
7.3	Hardware & Software Requirements	
8	**Methodology**	24
8.1	Spiral Model	
8.2	Objectives of SpiralModel and its advantages	
9	**Operating Tools and Development Environment**	28
9.1	Frontend Development tools	
9.2	Development Environment	
10	**Designing**	32
10.1	Activity Diagrams	
10.2	Entity Relationship Diagram	
10.3	Sequence Diagrams	
11	**Implementation**	45
11.1	Screenshots of projects	
12	**Testing**	52
12.1	Unit Testing	
12.2	System Testing	
12.3	Operational Testing	
12.4	Maintenance	
13	**Reference**	57

Chapter 1

1. ABSTRACT

Resume Builder+ is a web application developed to assist users in creating professional resumes quickly and effortlessly. In today's competitive environment, job seekers often face challenges in crafting a resume that effectively showcases their skills, qualifications, and achievements. Resume Builder+ addresses this need by providing a streamlined platform where users can input their information and generate a well-structured, visually appealing resume.

The platform features an intuitive user interface that guides individuals through various sections of resume building such as personal information, education, work experience, and skills. Users can preview their resumes in real time and customize them using pre-designed templates, ensuring a personalized and polished final output.

The application is developed using HTML, CSS, and JavaScript on the front end to ensure an interactive and responsive user experience. Instead, the system utilizes file-based or session-based storage to manage user input efficiently, keeping the application lightweight and suitable for local or small-scale deployments.

The expected outcomes of this project include improving the ease and speed of resume creation, reducing design-related barriers for users, and enhancing the overall quality of resumes. With Resume Builder+, the goal is to make resume writing more accessible, convenient, and user-friendly for all.

1.1 Overview

In today's fast-paced digital world, job seekers often face challenges in designing professional resumes that effectively present their skills and experience. **Resume Builder+** was created to solve this problem by offering an easy-to-use platform that guides users through the resume creation process. Whether someone is applying for their first job or looking to improve their existing resume, this platform simplifies the task by providing customizable templates and real-time previews—making resume building accessible to everyone, regardless of their technical knowledge.

1.2 Objective

To provide a simple and user-friendly platform for resume creation.

- To allow users to enter their personal, academic, and professional details through guided forms.
- To enable customization of resumes with design templates and layout options.
- To offer real-time preview and downloadable resumes in print-ready formats.

.

1.3 Purpose, Scope, and Applicability

1.3.1 Purpose

- Many individuals find it difficult to design resumes that look professional. This platform addresses that issue by simplifying the design process.

- Resume Builder+ reduces dependency on third-party tools by offering an all-in-one resume builder in a web format.

- The platform also empowers users to focus on content rather than formatting, increasing productivity and confidence.

1.3.2 Scope

- Resume Builder+ includes features such as guided data entry forms, live preview, multiple templates, and file download options.

- The application uses **HTML, CSS, and JavaScript** for the frontend.

- It is best suited for students, job seekers, and professionals who need to create or update resumes quickly and effectively.

1.3.3 Applicability

- Helpful for individuals applying for internships, jobs, or academic opportunities.
- Useful in educational institutions for students to create resumes during placement drives.
- Can be used offline or in local networks for workshops or training sessions due to its lightweight nature.

1.4 Achievements

With Resume Builder+, users are able to create customized, print-ready resumes in minutes. The platform eliminates the need for complex design software or online subscriptions, giving users complete control over their resume structure and appearance. It promotes independence and boosts confidence among job seekers.

1.5 Organization of Report

This report outlines the design and development of the Resume Builder+ platform, highlighting the integration of front-end and back-end technologies to deliver a seamless user experience. It details the functionality, modules, and benefits for end users, particularly students and job seekers.

Chapter 2

1 LITERATURE SURVEY

In recent years, several web-based resume-building platforms have been developed to simplify the process of creating professional resumes. An analysis of existing systems reveals a range of benefits as well as key limitations that affect usability and accessibility.

Popular platforms such as **Canva**, **Zety**, and **Novoresume** offer modern design templates and interactive guidance. However, many of these tools require user registration, paid subscriptions for full access, or depend heavily on online databases and cloud services, which may not be ideal for users with limited internet access or financial constraints.

Moreover, research in **web usability and accessibility** suggests that complex platforms with too many features can overwhelm users, especially students or first-time job seekers. A minimalist, guided approach is often more effective for creating structured resumes with ease.

Resume Builder+ addresses these challenges by offering a simple, fully client-driven resume builder that uses **HTML, CSS, and JavaScript** for the frontend .

Chapter 3

3. Acquisition of Knowledge

Before initiating the development of **Resume Builder+**, a thorough knowledge acquisition phase was carried out to ensure that the final product would be simple, efficient, and accessible to a wide range of users. This phase focused on understanding user needs, market trends, and the technical skills required to implement a fully functional resume-building platform using lightweight web technologies.

3.1 Understanding the Domain

To develop an effective resume builder, it was important to explore:

- The structure and content of modern resumes for different industries.
- The difficulties users often face, such as formatting issues or lack of design guidance.
- The expectations of first-time users, especially students and job seekers creating resumes for internships or entry-level positions.

3.2 Market Research

Popular resume platforms like **Canva**, **Zety**, and **VisualCV** were analyzed to identify:

- Common features such as template selection, real-time editing, and export functionality.
- Usability issues, such as limited free access, account requirements, and complex interfaces.
- Gaps that ResumeBuilder+ could address, including offline compatibility, simplicity, and full feature access without hidden charges.

3.3 Technical Knowledge

Key technologies were selected and studied to align with the project's goals:

- **Frontend**: HTML, CSS, and JavaScript were used to design an interactive and user-friendly interface for resume creation.

3.4 Functional Integration Study

Special focus was given to:

- Creating downloadable resume templates in PDF and print cv.
- Managing user inputs in a secure and organized way without a database.
- Ensuring compatibility across browsers and devices for smooth usage.

3.5 Legal and Ethical Considerations

While the platform does not handle sensitive data, the following principles were considered:

- Respect for user privacy by avoiding unnecessary data storage.
- Clear, simple design to ensure accessibility for all users, including those with limited technical skills.

3.6 Skill Building

The development team enhanced their skills through:

Practicing by building mini projects and components (e.g., form handling, file generation).

Outcome of Knowledge Acquisition

- Clear technical goals were established based on user needs.
- A lightweight, database-free architecture was chosen to simplify deployment and improve performance.
- The team developed a practical skillset to ensure smooth development and user-friendly output.
- ResumeBuilder+ was shaped to be minimal, effective, and accessible to everyone.

Chapter 4

4. Domain Knowledge

4.1 Introduction to the Domain

The ResumeBuilder+ project operates within the domain of Digital Career Development Tools. This domain focuses on using web technologies to assist individuals in crafting professional documents such as resumes and cover letters, essential for job applications. As digital hiring processes continue to dominate, the need for accessible, personalized, and easy-to-use resume-building platforms has increased significantly.

Online resume builders provide an efficient alternative to manual formatting, enabling users to quickly create well-structured resumes without needing advanced design skills or word processing software. These platforms combine web design, document generation, and user interaction to enhance job-seeking experiences.

4.2 Key Components of the Domain

Resume Creation Tools

These are the core of any online resume builder. They allow users to enter personal, academic, and professional information, which is then formatted into a predefined layout for consistency and visual appeal.

Template Systems

Pre-designed resume templates guide users in organizing their content. Templates ensure alignment with professional standards and enhance readability.

Real-Time Editing

Interactive forms built with JavaScript allow users to instantly see updates or preview their resume layout as they enter information.

Export Options (PDF/Word)

The ability to export resumes in professional formats (typically PDF or DOC) is a key feature.

Offline/Lightweight Functionality

Unlike complex SaaS platforms, ResumeBuilder+ avoids the need for external databases, enabling faster load times and reduced reliance on server storage—ideal for users with limited internet access or basic devices.

User Privacy and Simplicity

By not storing user data, the platform prioritizes privacy. No login is required, making it ideal for first-time users or those concerned about data sharing.

4.3 Importance of Domain Knowledge in Project Development

User-Centric Design

Understanding how users think and behave while building resumes allows for an intuitive interface, with logical field grouping (e.g., personal info, education, experience), and simple navigation.

Content Structuring

Domain knowledge ensures that the resume sections and content types provided (skills, summary, achievements) meet industry expectations and recruiter preferences.

Performance Optimization

With insights from this domain, the system is built to function efficiently on various devices and browsers, without heavy resource usage.

Accessibility and Relevance

Domain familiarity ensures that features remain accessible to all users—whether they are students, job seekers, or professionals making quick edits before interviews.

4.4 Advantages of Domain Knowledge

- Allows for the creation of a resume-focused, user-friendly interface tailored to job seekers.
- Helps anticipate real-world user behavior, such as resume format preferences and frequent edits.
- Enhances platform usability, especially for non-technical users.
- Guides design decisions such as font choices, color schemes, and section layout for professional appeal.
- Enables rapid development of features that match user expectations without overcomplicating the experience.

Chapter 5

5. SYSTEM STUDY

The system study for the **ResumeBuilder+** project involved a thorough analysis of current methods job seekers use to create resumes, identifying common challenges and exploring how a digital platform can streamline the resume-building process. This study helped define key requirements, understand existing solutions, and outline how the proposed system would offer a simpler, more effective alternative.

5.1 Existing System

In the traditional approach, users often rely on:

- **Manual resume creation** using software like Microsoft Word or Google Docs, which requires formatting knowledge.
- **Online templates** that are either difficult to customize or require account creation and payment.
- **Job portals** that offer resume tools but have limited export or design options.

Common limitations of these systems include:

- Lack of **real-time preview** while editing.
- No **interactive templates** or design customization.
- Dependence on **external storage** or account-based data saving.
- Limited **control over personal data**, especially when resumes are stored online.

These shortcomings create barriers, especially for non-technical users, and slow down the process of resume creation.

5.2 Proposed System

The **ResumeBuilder+** system addresses these issues by offering a lightweight, fully client-side resume-building platform powered by **HTML, CSS, JavaScript**, **PDF for generation**.

- **No login or data storage** – Users can build and export resumes without creating an account.
- **Real-time form-based editing** – Users fill out forms and instantly see a live preview.
- **Professional templates** – Clean and modern designs suitable for all professions.
- **Downloadable PDF** – Resumes are generated dynamically using PHP for export.
- **Mobile-friendly UI** – Ensures a smooth experience across devices.

5.3 Benefits of the Proposed System

- **Quick and accessible** – Users can generate resumes in minutes without technical skills.
- **No data stored** – Enhances user privacy and reduces security risks.
- **Free and offline-capable** – No subscription or internet dependency beyond resume generation.
- **User-friendly interface** – Minimalistic design focused on usability.
- **Responsive** – Works smoothly across phones, tablets, and desktops.

Chapter 6

6. PROBLEM DEFINITION & SCOPE OF PROJECT

Problem Definition

In today's digital world, many job seekers still struggle with creating professional resumes due to:

- Lack of technical skills in formatting or design.
- Dependence on expensive or subscription-based online platforms.
- Difficulty finding modern, customizable templates.
- Inability to preview resumes in real-time before downloading.
- Lack of a unified platform for building, previewing, and exporting resumes without account creation or data sharing.

There is no simple and secure system where users can:

- Enter resume information easily.
- Choose from modern resume templates.
- Preview their resume in real-time.
- Export the final version as a PDF.
- All while keeping their data private.

Objective

- To provide a **user-friendly platform** that enables users to quickly build professional resumes.
- To allow users to **edit, preview, and download** their resumes in real time.
- To offer **stylish, responsive templates** suitable for different job roles and industries.
- To **eliminate the need for database storage** or user logins to ensure privacy.
- To develop a solution using **HTML, CSS, JavaScript**.

Purpose

- To help users **create professional resumes without design skills** or expensive software.
- To reduce the barriers job seekers face by offering a **free, secure, and accessible resume-building platform**.
- To enable fast resume generation without requiring personal data storage or cloud syncing.
- To provide an **offline-capable tool** that works even with limited internet access (except during PDF export).

Scope

- The platform will include features such as:
    - Interactive forms for resume content input (personal info, education, experience, skills).
    - Responsive layout for mobile and desktop users.
- The system will focus on:
    - Simplicity and accessibility for non-technical users.
    - Ensuring complete **user data privacy** by avoiding account creation and database use.
    - Easy customization and download of resumes with minimal steps.

Chapter 7

7. Requirement Analysis

The requirement analysis phase is crucial to understanding what the system should do and how it should perform to meet the needs of its users.

Analysis

The Resume Builder website is aimed at simplifying the process of creating professional resumes for job seekers. The system is designed to address common challenges users face, such as lack of design skills, formatting issues, and difficulty in exporting clean resumes. Based on user behavior and trends, the following insights were derived:

- Users prefer guided resume creation rather than building from scratch.
- A clean and professional template is essential.
- Simple user interface increases usability for all age groups.
- Output in downloadable format (PDF) is a key requirement.

Feasibility Study

1. Technical Feasibility:
    - Frontend will use well-supported web technologies: HTML, CSS, JavaScript.
    - Resume export as PDF can be done using libraries like TCPDF or Dompdf.
2. Operational Feasibility:
    - Platform is designed to be lightweight and simple.
    - Minimal learning curve for users to create resumes.
    - Users can complete a resume in a few easy steps.
3. Economic Feasibility:
    - Low-cost development using open-source tools.
    - No need for expensive hosting or server resources.

    .
4. Legal Feasibility:
    - Since no user data is stored in a database, privacy compliance is simpler.
    - If session or cookie data is used, user consent may be included.

Hardware & Software Requirements

Hardware Requirements:

For End Users:

- Smartphone, Tablet, or PC
- Internet connection
  For Admin/Hosting:
- Basic server with:
  - Processor: Intel i3 or equivalent
  - RAM: 2GB minimum
  - SSD preferred for performance

Software Requirements:

- Frontend: HTML, CSS, JavaScript
- PDF Export: Dompdf / TCPDF (PHP-based libraries)
- IDE/Editor: VS Code, Sublime Text, or any code editor
- Browser Support: Chrome, Firefox, Edge, Safari (latest versions)

Chapter 8

8.1 Spiral Model

For the development of the **Resume Builder Website**, the **Spiral Model** was adopted due to its iterative and flexible nature. This approach supports continuous improvements and risk management, which is ideal for a system that focuses heavily on user experience, real-time editing, and printable output. Since users interact dynamically with resume templates, the Spiral Model helped ensure features were tested, refined, and improved through repeated cycles.

Each loop of the spiral represents a distinct development phase, helping identify risks, validate features, and refine the product progressively.

8.2 Objective of Spiral Model in Resume Builder

Each loop in the Spiral Model was mapped to a development phase for the Resume Builder, enabling changes based on feedback and technical considerations.

1. Determining Objectives, Alternatives, and Constraints:

In the first phase, the primary goals were defined:

- Allowing users to create and preview resumes online.
- Offering multiple layout templates with styling options.
- Making the site user-friendly without requiring any login or data storage.
- Using only HTML, CSS, JavaScript (frontend).

2. Risk Analysis and Evaluation of Alternatives:

Risks identified during planning:

- Managing user data without a database.
- Ensuring form data persists during the session (without losing content on refresh or navigation).
- Printing or exporting resumes while maintaining formatting.

Mitigation strategies included:

- Using JavaScript to manage dynamic updates and preview changes in real-time.
- Exporting resumes as print-friendly HTML pages or integrating basic PDF generation libraries.

3. Engineering (Design and Development):

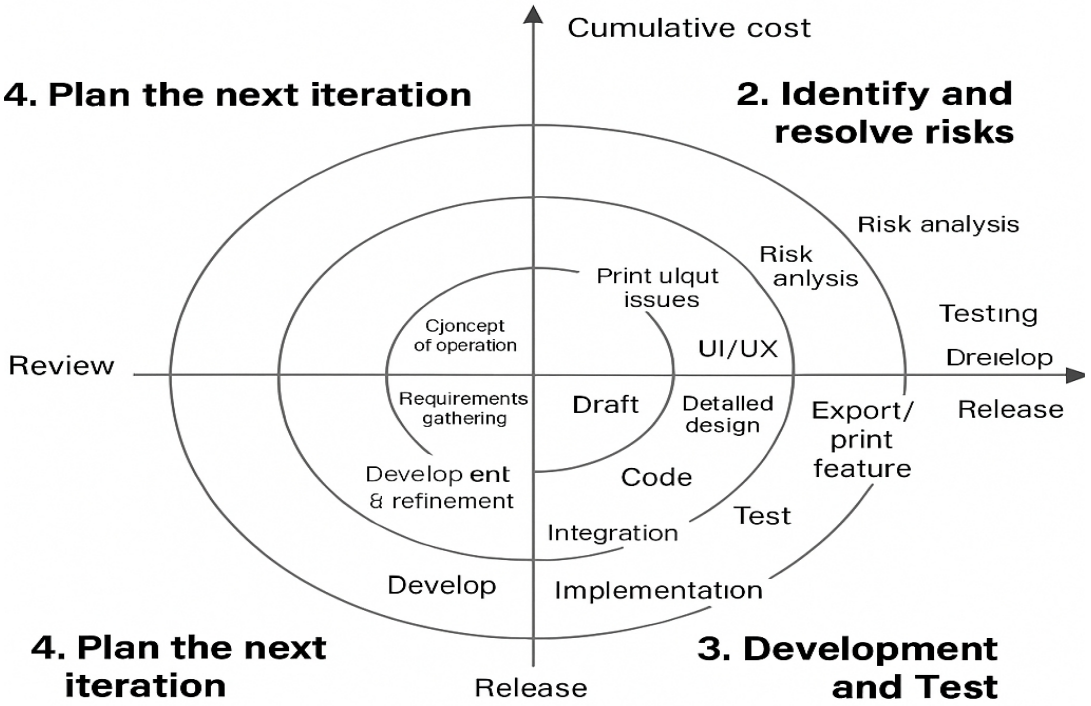
Development was carried out in iterations:

- **Phase 1:** Form layout for user input (name, education, experience).
- **Phase 2:** Live preview using JavaScript.
- **Phase 3:** Style customization with multiple templates (CSS themes).
- **Phase 4:** Download/Print feature using window.print() or HTML-to-PDF tools.

Each cycle included user testing and enhancements based on usability.

Spiral Model Phases Applied to Resume Builder

- **Planning:** Features like live preview, section editing, and print/download were outlined.
- **Risk Analysis:** Focused on managing data temporarily without a database, and maintaining layout consistency during export.
- **Engineering:** Used HTML, CSS, and JavaScript to build UI and logic.
- **Evaluation:** Iterative user feedback helped fine-tune resume layouts and simplify user interaction.



Resume Builder

Chapter 9

9. Operating Tools and Development Environment

The development of Resume builder+ required a modern tech stack that ensured smooth user experience, robust backend processing, and secure handling of sensitive health-related data. Below is a comprehensive breakdown of all tools and technologies used in the project:

Frontend Development Tools

1. HTML

- HTML (HyperText Markup Language) is the standard language used to create and structure web pages.
- It is platform-independent and supported by all web browsers.
- HTML allows for embedding images, videos, and audio into web pages.
- It supports hyperlinks, enabling navigation between different pages.
- HTML is a markup language that provides the structure and layout of a website.
- HTML can be integrated with CSS and JavaScript to enhance the user experience.
- HTML5 introduced new features, including support for geolocation, offline storage, and semantic elements.

2. CSS

- CSS3 was used to style the user interface, including colors, spacing, fonts, and layout alignment.
- Enabled media queries to make the web application responsive across devices (mobile, tablet, desktop).
- Used for animations and transitions to enhance interactivity, such as button effects and pop-up modals.
- CSS Grid and Flexbox layouts allowed the site to be visually structured and adaptable.

3. JavaScript

- JavaScript is a versatile programming language used to add interactivity to web pages.
- It runs on the client side, enabling dynamic updates without reloading the page.
- JavaScript can manipulate HTML and CSS to create dynamic content like forms, animations, and more.
- It supports various frameworks and libraries like React, Angular, and Vue for building complex frontends.
- JavaScript has a rich set of APIs for tasks like handling events, fetching data, and creating visual effects.
- JavaScript improves user experience by enabling fast and interactive features like sliders, modals, and dropdowns.

Operating Environment

5. Operating Systems (Windows, macOS, Linux)

- Supported all development tools and technologies used.
- Enabled running MongoDB, Node.js, and VS Code without compatibility issues.
- Provided built-in tools like PowerShell or Bash for executing scripts and commands.
- Linux/Ubuntu preferred for hosting and production environment due to its stability and performance.
- Compatibility ensured a seamless development experience across different team members' devices.

6. Browsers (Chrome, Firefox, Safari)

- Regularly tested the platform across browsers to ensure cross-browser compatibility.
- Utilized built-in developer tools for inspecting UI, console errors, and network behavior.
- Verified that responsive design and interactive elements work uniformly on each browser.
- Used in manual and automated testing for real-time performance checks.
- Allowed features like WebRTC (used in video calling) to function consistently across platforms.

Chapter 10

10. SYSTEM DESIGN

10.1 Activity Diagrams.

An **activity diagram** is a type of **UML (Unified Modeling Language)** diagram that visually represents the **workflow** or **activities** involved in a process or system. It's commonly used in software development, business process modeling, and system analysis.

Purpose:

To show **how tasks flow** from one to another, **decision points**, **parallel processes**, and how the process **starts and ends**.

Key Elements:

Initial State or Start Point:

● A small filled circle followed by an arrow represents the initial action state or the start for any activity diagram.

● For activity diagram using swim lanes, make sure the start point is placed in the top left corner of the first column.

Start Point/Initial State

Activity or Action State:

● An action state represents the non-interruptible action of objects.

● You can draw an action state in Smart Draw using a rectangle with rounded corners.

Activity

Action flow:

● Action flows, also called edges and paths, illustrate the transitions from one action state

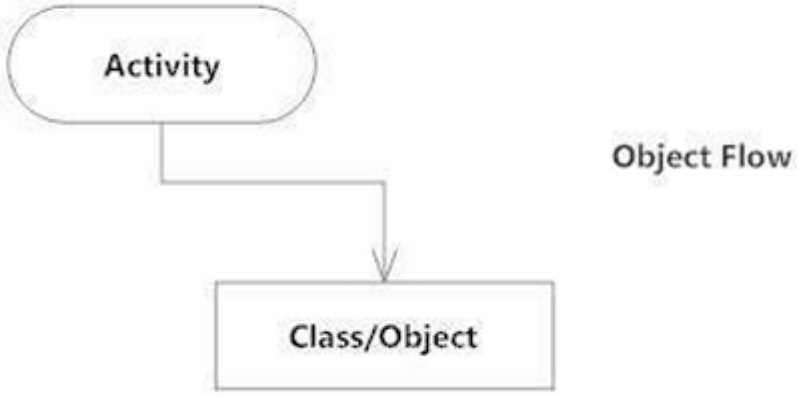
● They are usually drawn with an arrowed line.

Object Flow:

● Object flow refers to the creation and modification of objects by activities.

● An object flow arrow from an action to an object means that the action creates or influences the object.

● An object flow arrow from an object to an action indicates that the action state uses the object.



Decisions and Branching:

● A diamond represents a decision with alternate paths.

● When an activity requires a decision prior to moving on to the next activity, add a diamond between the two activities.

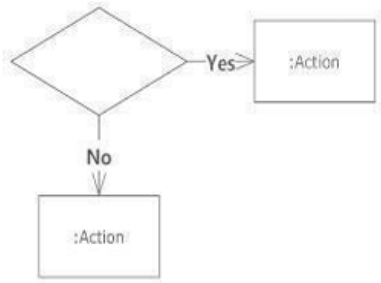
● The outgoing alternates should be labelled with a condition or guard expression.

● You can also label one of the paths “else”.

Guards:

● In UML, guards are a statement written next to a decision diamond that must be true before moving next to the next activity.

● These are not essential, but are useful when a specific answer, such as "Yes, three labels are printed" is needed before moving forward.



Guard Symbols

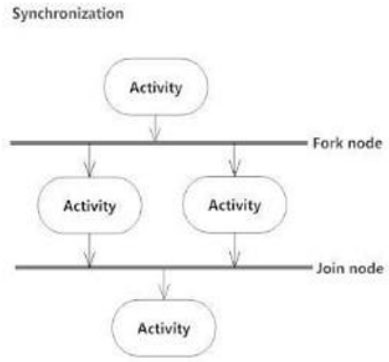
Synchronization:

● A fork node is used to split a single incoming flow into multiple concurrent flows.

● It is represented as a straight, slightly thicker line in an activity diagram.

● A join node joins multiple concurrent flows back into a single outgoing flow.

● A fork and join mode used together are often referred to asynchronization.

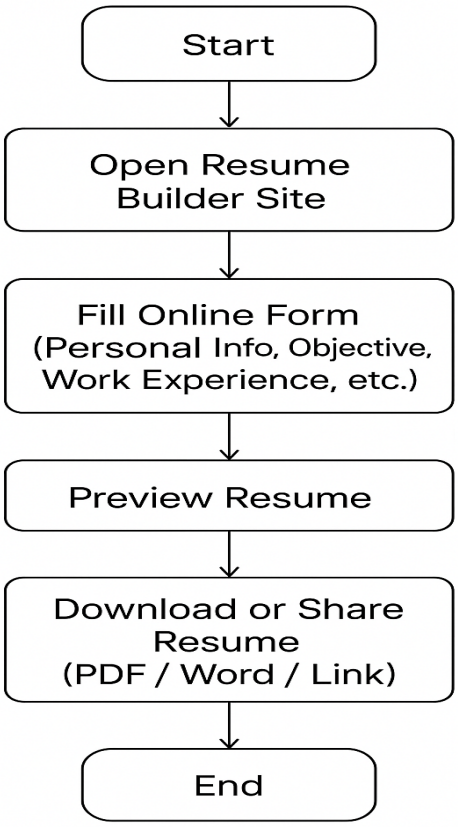


Final State or End Point:

● An arrow pointing to a filled circle nested inside another circle represents the final action state.

End Point Symbol

Activity Diagram Of Online Resume Builder



10.2 ENTITY RELATIONSHIP DIAGRAM

✓ What is an Entity Relationship Diagram (ERD)?

● An ERD shows the relationships of entity sets stored in a DB.

● An entity in this context is a component of data. In other words, ER diagrams illustrate the

● logical structure of databases.

● At first glance an entity relationship diagram looks very much like a flowchart.

● It is the specialized symbols, and the meanings of those symbols, that make it unique.

Common Entity Relationship Diagram Symbols:

● An ER diagram is a means of visualizing how the information a system produces is relating.

● There are five main components of an ERD:

Entities:

● An entity is represented by rectangle. An entity is an object or concept about which you want to store
information.

Weak Entities:

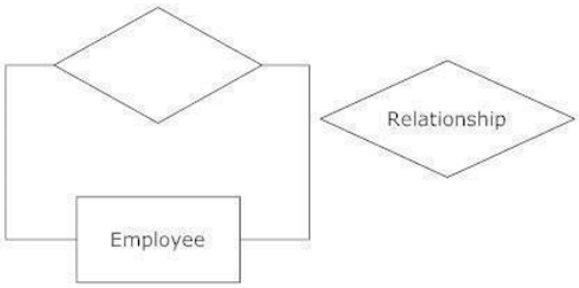
● A weak entity is an entity that must defined by a foreign key relationship with another entity as it cannot
be uniquely identified by alone.

✓ Actions:

● Actions which are represented by diamond shapes, show how two entities share information
in the database.

● In some cases, entities can be self-linked.

● For example, employees can supervise other employees.



Attributes:

- Attribute which are represented by ovals.
- A key attribute is the unique, distinguishing characteristic of the entity.
- For example, an employee's social security number might be the employee's key attribute.

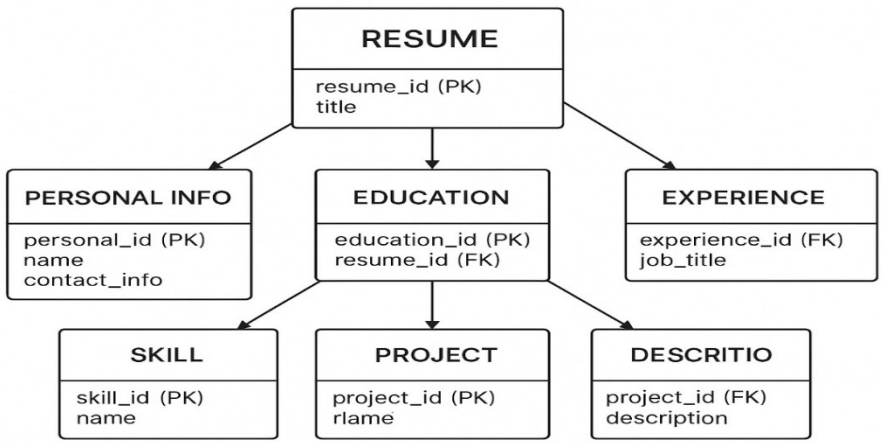
A **multivalued** attribute can have more than one value. For example, an employee entity can have multiple skill values.

A **derived attribute** is based on another attribute. For example, an employee's monthly salary is based on the employee's annual salary.

Connecting lines:

● Solid lines that connect attributes to show the relationships of entities in the diagram.

Entity Relationship Diagram Online Resume Builder



10.3 SEQUENCE DIAGRAMS: -

✓ What is a Sequence Diagram?

● Sequence diagrams describe interactions among classes in terms of an exchange of messages over

time. They're also called event diagrams.

● A sequence diagram is a good way to visualize and validate various runtime scenarios.

● These can help to predict how a system will behave and to discover responsibilities a class may need

to have in the process of modelling a new system.

✓ Basic Sequence Diagram Notations: -

✓ Class Roles or Participants:

● Class roles describe the way an object will behave in context. Use the UML object symbol to illustrate

class roles, but don't list object attributes.

Activation or Execution Occurrence:

● Activation boxes represent the time an object needs to complete a task.

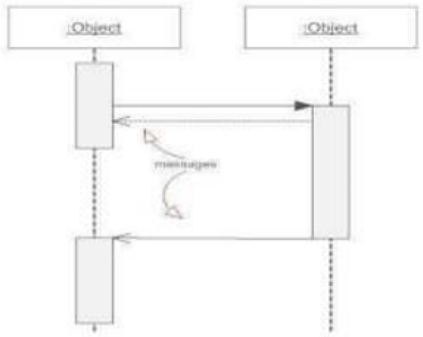
● When an object is busy executing a process or waiting for a reply message, use a thin gray rectangle placed vertically on its lifeline

Messages:

● Messages are arrows that represent communication between objects.

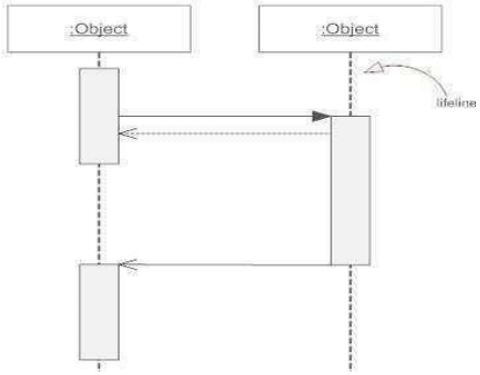
● Use half-arrowed lines to represent asynchronous messages.

● Asynchronous messages are sent from an object that will not wait for a response from the receiver before continuing its tasks.



Lifelines:

● Lifelines are vertical dashed lines that indicate the object's presence overtime.



Loops:

● A repetition or loop within a sequence diagram is depicted as a rectangle. Place the condition for exiting

the loop at the bottom left corner in square brackets [].

Types of Messages in Sequence Diagrams: -

o Synchronous Message:

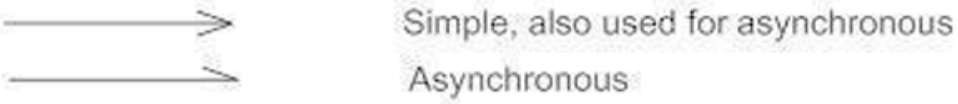
● A synchronous message requires a response before the interaction can continue.

● It's usually drawn using a line with a solid arrowhead pointing from one object to another.

Asynchronous Message:

● Asynchronous messages don't need a reply for interaction to continue.

● Like synchronous messages, they are draw with an arrow connecting two lifelines; however, the arrowhead is usually open and there's no return message depicted.

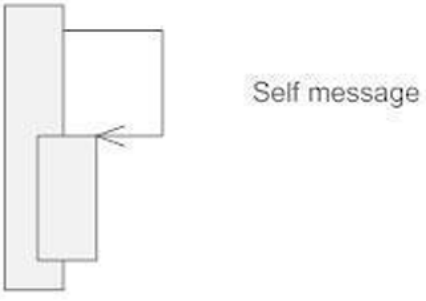


o Reply or Return Message:

● A reply message is drawn with a dotted line and an open arrowhead pointing back to the original lifeline.

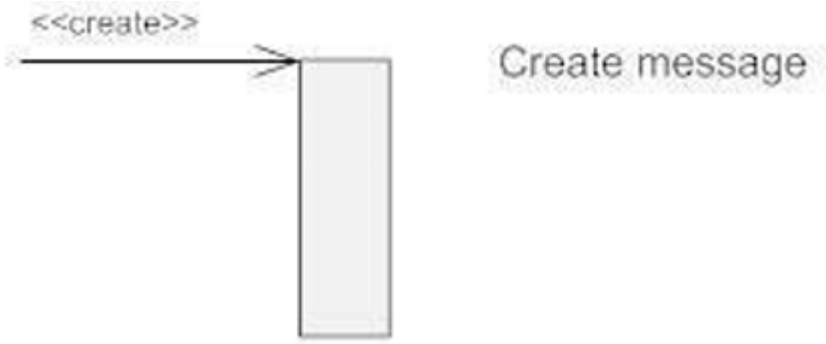
Self-Message:

● A message an object sends to itself, usually shown as a U-shaped arrow pointing back to itself.



Create Message:

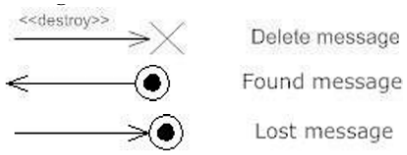
● This is a message that creates a new object. Similar to a return message, it's depicted with a dashed line and an open arrowhead that points to rectangle representing the object created.



Delete Message:

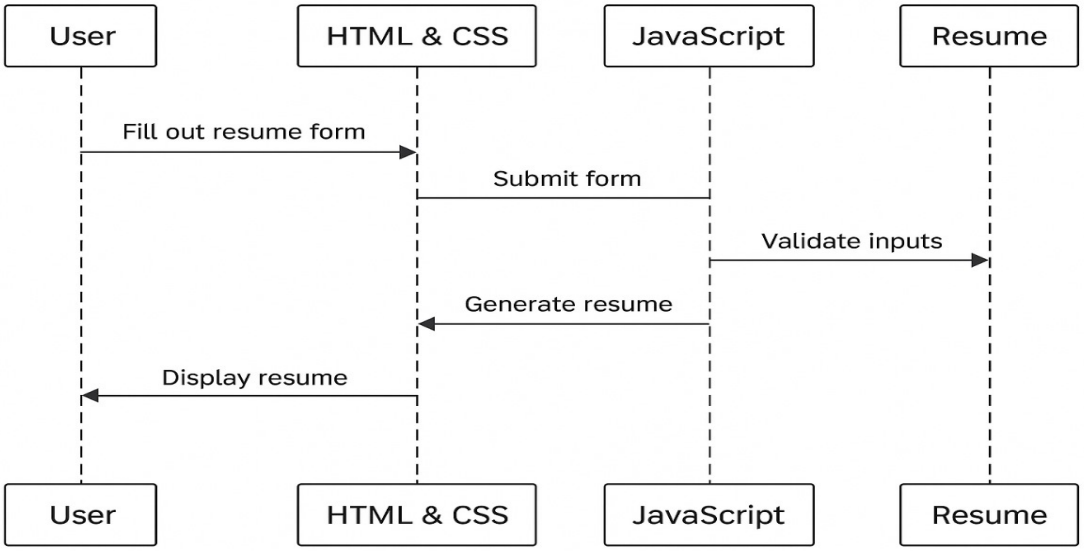
● This is a message that destroys an object. It can be shown by an arrow with an x at the end.

● Found/Lost Message:



● A message sent from/to an unknown recipient, show by an arrow from an endpoint to a line

Sequence Diagrams Online Resume Builder



Chapter 11

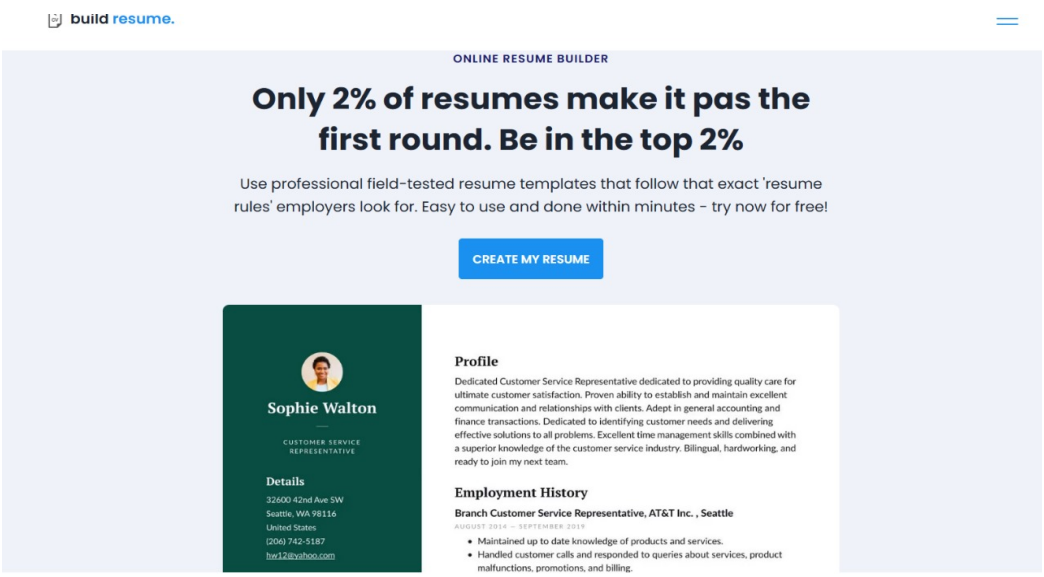
11 Implementation

11.1 Screenshots

Homepage

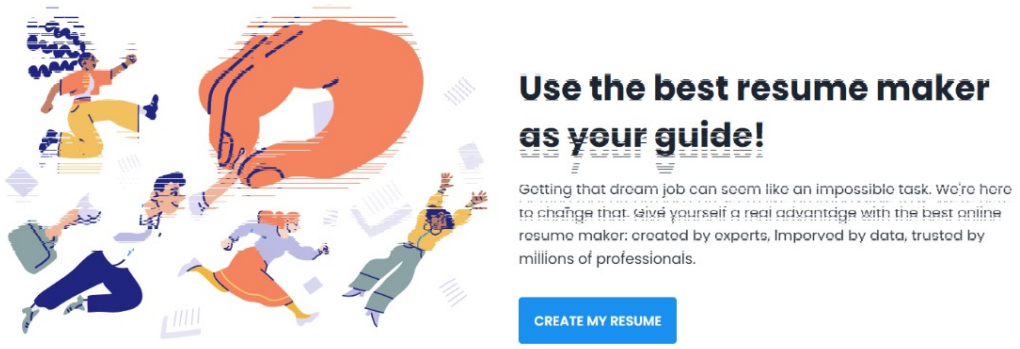
The homepage of the **"Build Resume"** website presents a sleek, modern interface designed to help job seekers create standout resumes quickly and effectively. At the top, a clean navigation bar displays the platform's name.The main section features a minimalist design with bold typography, highlighting the key benefit: **"Only 2% of resumes make it past the first round. Be in the top 2%"**.A short paragraph beneath this message reassures users that the platform uses professionally designed, field-tested templates tailored to what hiring managers look for. It emphasizes ease of use, encouraging users to take action right away.A prominent **"Create My Resume"** button invites visitors to start building their professional resume in minutes.To visually support this, a polished resume preview is displayed beside the call to action, showcasing what users can achieve with the tool. The sample includes a photo, name, title, contact details, profile summary, and work experience, all styled cleanly for maximum impact.This homepage balances functionality with inspiration, giving users the confidence to start building their future, one resume at a time.

.



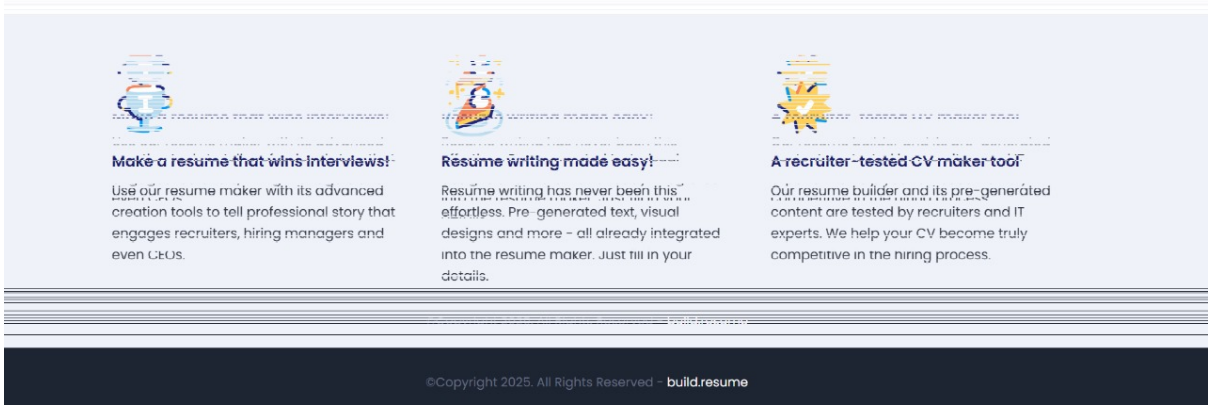
Review

Your homepage immediately communicates what the product is a resume builder designed to help users create high-quality, professional resumes that stand out.
The headline, *"Only 2% of resumes make it past the first round. Be in the top 2%"*, is bold, attention-grabbing, and plays on a real pain point many job seekers feel. CREATE MY RESUME" is action-driven and clearly visible.The button is well-placed and stands out visually.

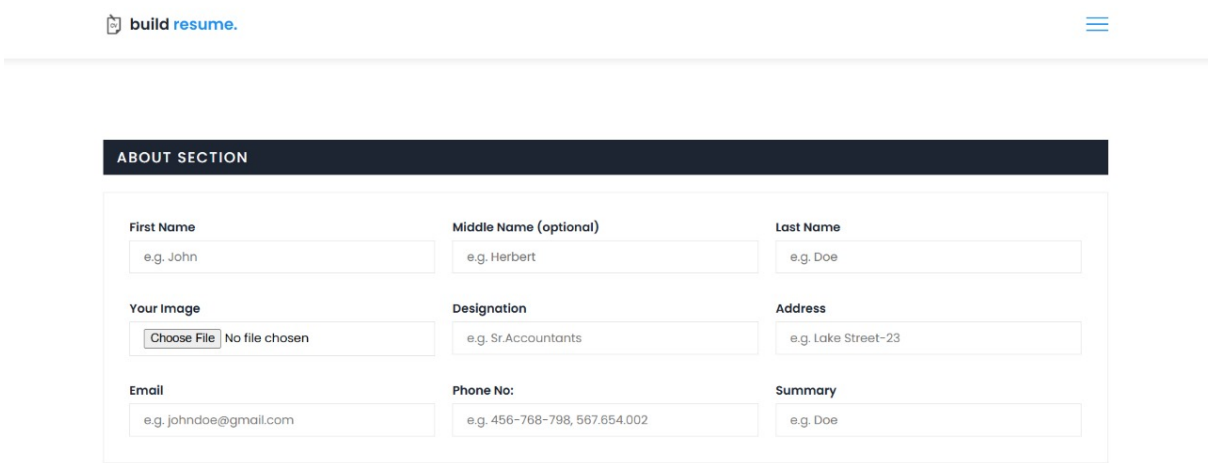


About Section

A large stylized hand holding a pen, symbolizing resume writingDiverse animated figures running or jumping, likely symbolizing job seekers in pursuit of their goals., Getting that dream job can seem like an impossible task. We're here to change that. Give yourself a real advantage with the best online resume maker: created by experts, improved by data, trusted by millions of professionals. This image is likely part of a landing page or advertisemen encouraging users to use a professional resume builder to improve their job prospects.



Promotes the use of advanced resume creation tools that help craft a professional story. These tools are designed to capture the attention of recruiters, hiring managers, and even CEOs. Emphasizes ease of use. Offers features like pre-generated text and visual designs that are built into the tool—users just need to input their personal details. This section is meant to reassure potential users of the resume builder's effectiveness, ease of use, and credibility in the job market. This image is likely part of a landing page or advertisement encouraging users to use a professional resume builder to improve their job prospects.



The image you've uploaded shows the "ABOUT SECTION" of an online resume builder. It contains input fields for entering personal and professional information. Here's a breakdown of the fields available:

First Name – Example: John Middle Name (optional) – Example: Herbert

Last Name – Example: Doe

Your Image – Option to upload a photo file

Designation – Example: Sr. Accountants

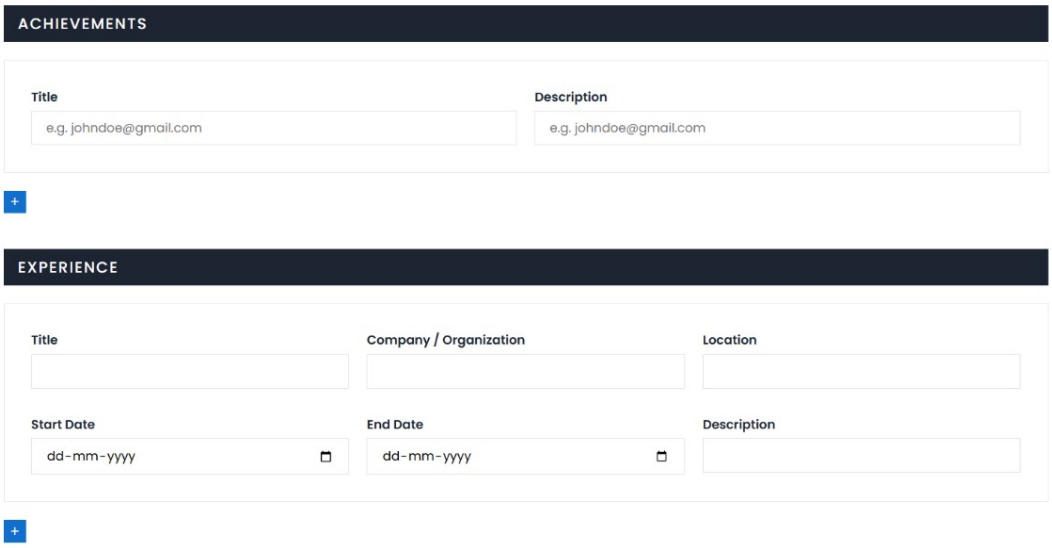
Address – Example: Lake Street-23

Email – Example: johndoe@gmail.com

Phone No. – Example: 456-768-798 or 567.654.002

Summary – Example: A short professional summary or description

Let me know if you'd like help filling this out or generating a complete resume from this!



Achievements and Experience in the resume builder

Achievements Section

Title – The name of the achievement (e.g., “Best Employee Award”)

Description – Details or context about the achievement

There is also a "+" button to add more achievements.

Experience SectionTitle – Job title (e.g., Software Developer)

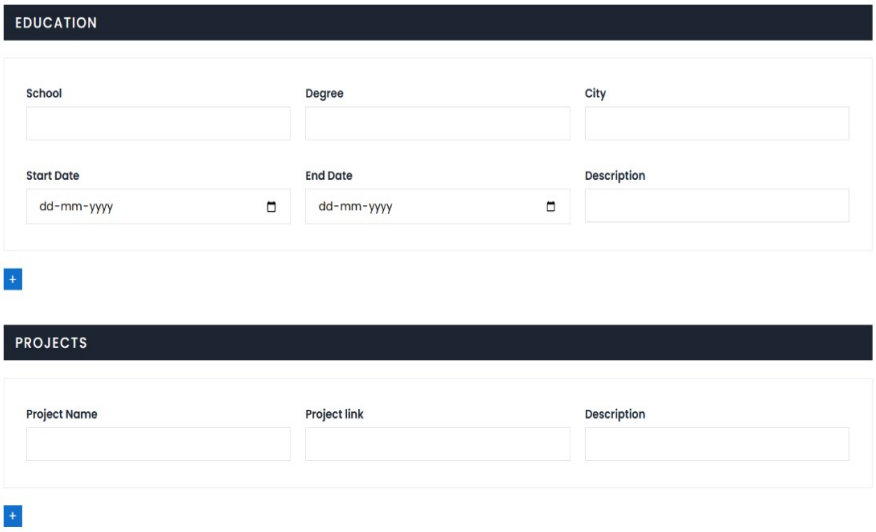
Company / Organization – Name of the employer

Location – Where the job was based

Start Date and End Date – Job duration (dd-mm-yyyy format)

Description – A brief about responsibilities, accomplishments, or role specifics

Another "+" button here allows adding multiple job entries.



Degree Title – e.g., Bachelor of Science in Computer Science

Institution Name – e.g., University of Delhi

Location – e.g., New Delhi, India

Start Date and End Date – e.g., August 2018 – May 2022

Additional Details (optional) – e.g., CGPA: 8.5/10, Relevant Coursework: Data Structures, Machine Learning

You might also include:

Honors (like Magna Cum Laude)

Scholarships or academic awards

HTML5, CSS3, JavaScript, React.js, Node.js, Express.js, MongoDB

RESTful APIs, Git & GitHub, Responsive Design, Bootstrap

Agile Methodology, Debugging, Problem-Solving.

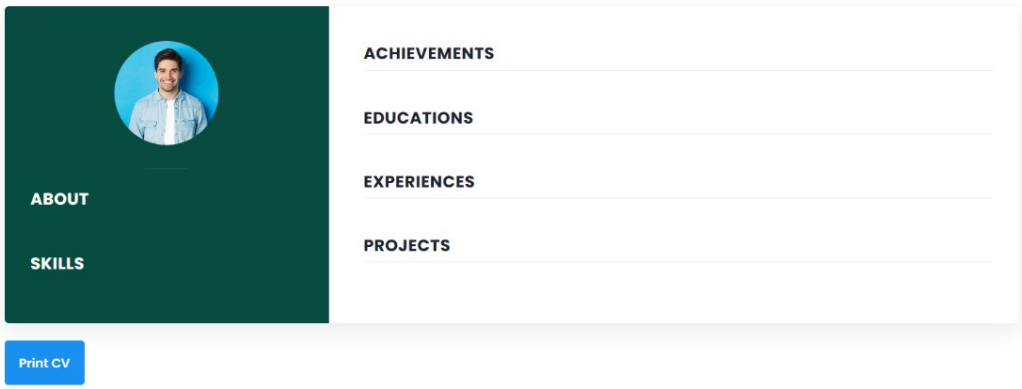
Showcase skills through your achievements and work experience:

Full Stack Developer Intern | Tech Solutions Inc.

Jan 2024 – Apr 2024

Developed a full-stack web application using React.js, Node.js, and MongoDB, improving load times by 30%.

Collaborated with a team using Agile practices and Git for version control.



Chapter 12

12 Testing

The process of testing was conducted using unit testing, integration testing, and operational testing.

12.1 Unit Testing (Developer)

Unit Testing is a level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

A unit is the smallest testable part of software. It usually has one or a few inputs and usually a single output. In procedural programming a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/super class, abstract class or derived/child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.)

Unit testing frameworks, drivers, stubs and mock or fake objects are used to assist in unit testing.

Method

Unit testing is performed by using the White Box Testing method.

When is it performed?

Unit testing is the first level of testing and is performed prior to Integration Testing.

Who performs it?

Unit Testing is normally performed by software developers themselves or their peers. In rare cases it may also be performed by independent software testers.

Tasks

- Unit Test Plan :- Prepare, Review, Rework, Baseline
- Unit Test Cases/Scripts :-Prepare, Review, Rework, Baseline
- Unit Test :- Perform

Benefits

- Unit testing increases confidence in changing/maintaining code. If good unit tests are written and if they are run every time any code is changed, the likelihood of any defects due to the change being promptly caught is very high. If unit testing is not in place, the most one can do is hope for the best and wait till the test results at higher levels of testing are out. Also, if codes are already made less interdependent to make unit testing possible, the unintended impact of changes to any code is less.
- Codes are more reusable. In order to make unit testing possible, codes need to be modular. This means that codes are easier to reuse.
- Development is faster. How? If you do not have unit testing in place, you write your code and perform that fuzzy 'developer test' (You set some breakpoints, fire up the GUI, provide a few inputs that hopefully hit your code and hope that you are all set.) In case you have unit testing in place, you write the test, code and run the tests. Writing tests takes time but the time is compensated by the time it takes to run the tests. The test runs take very less time: You need not fire up the GUI and provide all those inputs. And, of course, unit tests are more reliable than 'developer tests'. Development is faster in the long run too. How? The effort required to find and fix defects found during unit testing is peanuts in comparison to those found during system testing or acceptance testing.
- The cost of fixing a defect detected during unit testing is lesser in comparison to that of defects detected at higher levels. Compare the cost (time, effort, destruction, humiliation) of a defect detected during acceptance testing or say when the software is live.
- Debugging is easy. When a test fails, only the latest changes need to be debugged. With testing at higher levels, changes made over the span of several days/weeks/months need to be debugged.
- Codes are more reliable. Why? I think there is no need to explain this to a sane person.

12.2 System testing (Test Manager)

System Testing is a level of the software testing process where a complete, integrated system/software is tested.

The purpose of this test is to evaluate the system's compliance with the specified requirements.

Here we had tested the entire application. The reference document for this process was the requirements document and the goal was to see that if the application meets the requirements.

12.3 Operational testing

It was performed on the realistic data to demonstrate that the application is working satisfactorily. Testing here was done to focus on the external behavior of the system; internal logic of the program is not emphasized.

Test Cases:

Testing is crucial to ensure the system performs accurately, efficiently, and securely under different conditions. The following test cases validate key components of the Resume builder+ platform.

Test Case ID	Test Description	Steps to Reproduce	Expected Result	Status
TC001	Check if homepage loads correctly	Open `index.html` in browser	Navigation bar, hero, form, and preview area should be visible	✅ Pass
TC002	Validate form input updates resume preview	Enter name, job title, summary and click “Generate Resume”	Preview section should update with input values	✅ Pass
TC003	Test empty input handling	Leave all fields empty and click “Generate Resume”	Preview should show default placeholders or empty values	✅ Pass / ❌ (if validation needed)
TC004	Test special character handling in inputs	Enter `John <script>alert(1)</script>` into name field	Script should not execute; special characters should be escaped	✅ Secure
TC005	Mobile responsiveness	Open site on mobile or reduce browser width	Layout should adapt (stacked layout, buttons & text visible)	✅ Pass
TC006	Button click triggers	Fill form, click “Generate Resume”	Page should not reload; preview	✅ Pass

Test Case ID	Test Description	Steps to Reproduce	Expected Result	Status
	JavaScript without reload		updates dynamically	
TC007	Test long input strings	Paste a very long name or summary	Page layout should stay intact, text should wrap or truncate nicely	✅ Pass / ❌ (style issue?)
C008	Resume preview reflects real-time updates	Change input after clicking generate	Preview updates immediately if implemented dynamically	✅ Pass / ❌ if not real-time

Maintenance

- **Website maintenance** for the Resume Builder project involves updating and enhancing the platform after deployment to ensure it remains functional, user-friendly, and aligned with evolving user needs.
- While maintenance is often seen as simply fixing bugs, in practice, a large portion (over 80%) of maintenance activities are focused on **adding new features, improving usability, and optimizing performance**. Many user-submitted issues are actually **requests for enhancements**, such as adding new templates or enabling export options like PDF/Word.
- Key **maintenance challenges** for the Resume Builder include both technical and managerial concerns:

  Keeping features aligned with user priorities (e.g., adding more modern templates, improving mobile responsiveness)

  Managing development resources — whether the updates are handled by the original dev team or passed to a maintenance team

  Accurately estimating update and feature implementation costs, including time for testing across browsers and devices

Chapter 13

REFRENCE

VisualCV: Online CV Builder & Professional Resume Maker

Resume Builder for 2025 | Free Resume Builder | Novorésumé

AI-Powered Job Scraping Tool & Labour Market Insights

Resume Help: Writing Services, Tips and Examples

Free Online Resume Builder - MyPerfectResume®