

Creating an **ATS-friendly resume** (Applicant Tracking System-friendly) means formatting your resume in a way that makes it easy for software to scan and extract your information.

✓ Step-by-Step Guide to Create an ATS-Friendly Resume

1. Choose a Simple and Clean Format

- **Use Word (.docx) or PDF** format (check the job posting to see if PDF is accepted).
- **Avoid tables, text boxes, graphics, images, and columns**, which ATS may not parse properly.
- **Use standard section headings** like: Professional Summary, Skills, Experience, Education.

2. Use a Clear and Readable Font

- Stick to common fonts: **Arial, Calibri, Times New Roman, Helvetica, Verdana.**
- Font size: 10–12 pt for body text, 14–16 pt for headings.

Text Type	Recommended Font Size
Body text (job details, bullet points)	11–12 pt ✓
Headings (e.g., "Professional Summary", "Experience")	13–14 pt ✓
Your Name at the Top	16–18 pt ✓
Contact Info	10–11 pt

3. Include Keywords from the Job Description

- Carefully read the job posting.
- Extract relevant **skills, tools, certifications, and qualifications**.
- Naturally include these keywords in your:
 - **Professional Summary**
 - **Skills**
 - **Work Experience**
 - **Projects**
 - **Education**

4. Structure Your Resume with Clear Sections

a.Contact Information

Name

Phone Number

Email

LinkedIn (optional)

Portfolio/GitHub (if applicable)

b.Professional Summary (2–4 lines)

A brief overview highlighting your title, experience, and key skills.

c.Skills

Use bullet points or a comma-separated list.

JavaScript, React, Node.js, Git, MongoDB, REST API, HTML, CSS

d.Professional Experience (most recent first)

For each job:

Job Title | Company | Location | Dates

- Action verb + responsibility + outcome/result

- Quantify when possible (e.g., Improved load time by 30%)

e.Education

Degree | Institution | Year | Location

f. Projects (if applicable)

Project Name

- Tools/technologies used
- What you built, how, and why it matters

5. Avoid Fancy Design Elements

 No:

- Graphics
- Logos
- Icons
- Colored backgrounds
- Charts

 Yes:

- Bullet points (•)
- Simple text formatting (bold for headings)

6. Save with Proper File Name

Use this format:

FirstName_LastName_JobTitle_Resume.docx

Example:

Kabir_Sagar_FrontendDeveloper_Resume.docx

7. Check with an ATS Scanner Tool

Use free online tools like:

- [Jobscan.co](https://www.jobscan.co)
- [Resumeworded.com](https://www.resumeworded.com)
- [SkillSyncer.com](https://www.skillsyncer.com)

These scan your resume against job descriptions and give optimization tips.

Example Summary (For MERN Developer):

Passionate MERN Stack Developer with 2+ years of experience building scalable web applications using React, Node.js, MongoDB, and Express. Skilled in REST API development and responsive UI design. Adept at Agile workflows and Git-based collaboration.

- Use **reverse chronological order** for jobs and education.
- No spelling or grammar mistakes — use Grammarly or similar tools.
- Always keep the **body text at least 11 pt**. Anything smaller is hard to read and may be skipped by some ATS software.

Accepted and ATS-Friendly Variants

You can safely use any of these:

- Professional Summary ✓
- Summary ✓
- Career Summary ✓
- Profile Summary ✓

These are standard section headings that ATS systems are programmed to recognize and parse.

Avoid These (Not ATS-friendly):

- ✗ About Me
- ✗ Introduction
- ✗ Overview
- ✗ Personal Profile

These might confuse older or simpler ATS software because they aren't standard resume section terms.

Sequence of Headings in Resume

1. Contact Information
2. Professional Summary
3. Skills
4. Projects

5. Education

6. Certifications / Trainings

7. Achievements / Activities

1. Customise Your Resume to the Job

- **Customize** your resume for each job you apply to.
 - Use **keywords** from the job description in your **summary, skills, and project sections**.
-

2. Highlight Projects Instead of Work Experience

- Include **college, personal, or freelance projects** that demonstrate your skills.
 - Clearly describe **what you built, which tools/technologies** you used, and **the outcome**.
-

3. Use a Strong Professional Summary

- Write a brief and focused **2–3 line introduction** at the top.
- Include your degree, career interest, and top skills.

 *Example:*

Computer Science graduate with hands-on experience in React and Node.js. Eager to join a fast-paced development team and contribute to real-world applications.

4. List Only Relevant Skills

- Avoid listing outdated or unrelated tools (like MS Paint or Notepad).
 - Be specific (e.g., **MongoDB** instead of **Database**).
-

5. Stick to a Simple and Clean Format

- Use **standard headings** (**Education**, **Projects**, **Skills**, etc.).
 - Avoid tables, multiple columns, graphics, or logos — they can break ATS parsing.
-

6. Use Action Words in Descriptions

- Start bullet points with verbs like: **Developed**, **Built**, **Implemented**, **Collaborated**, **Tested**, **Designed**.

Example:

- Built a MERN Stack e-commerce site with user authentication and payment gateway integration.
-

7. 🎯 Quantify Achievements Where Possible



- Even for projects or academic work, add numbers:
 - Built a website used by 200+ students
 - Improved API performance by 30%
-

8. 📁 Limit Resume to 1 Page

- As a fresher, **1 page is ideal**.
 - Focus on quality over quantity.
-

9. 🔍 Proofread and Eliminate Errors

- No typos, spelling, or grammar mistakes.
- Use tools like **Grammarly** or ask someone to review it. It means you should **carefully check** your resume for:
 - ✅ Spelling mistakes
 - ✅ Grammar errors
 - ✅ Punctuation issues
 - ✅ Wrong capitalizations

-  Inconsistent formatting
 -  Incorrect dates or contact details
-

10. Use the Right File Format and Naming

- Submit in **.docx** or **PDF** (based on employer's instruction).

Name your file professionally:

FirstName_LastName_Resume.pdf