A PROJECT REPORT ON

PERSONAL COMPUTER BUILDER

submitted in partial fulfillment of the

requirements for the award of degree

BACHELOR OF COMPUTER APPLICATIONS

Submitted by

SANDEEP B NAMBIAR (REG NO: 212102249)

PRITHIV RAJ K(REG NO: 212102247)

RITESH J.L (REG NO: 212102260)

Under the guidance of

Dr. A. CATHREEN GRACIAMARY., MCA., M.Phil., SET., NET., Ph.D.,

ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER APPLICATIONS

AGURCHAND MANMULL JAIN COLLEGE

(Affiliated to the University of Madras)

Chennai-600061

NOVEMBER 2023-APRIL 2024

AGURCHAND MANMULL JAIN COLLEGE

(Affiliated to the University of Madras)

Chennai-600061

BONAFIDE CERTIFICATE

Certified that this report titled **PERSONAL COMPUTER BUILDER** is a bonafide record to the of the project work done by **SANDEEP B NAMBIAR(REG NO: 212102249), PRITHIV RAJ K(REG NO:212102247), RITHESH J.L (REG NO: 212102260)** under the supervision and guidance, towards partial fulfillment of the requirement for award of the Degree of Bachelor of Computer Applications of Agurchand Manmull Jain College, Affiliated to the University of Madras during the year November 2023-April 2024.

Signature of Guide Signature of the Facilitator

DR.A.CATHREEN GRACIAMARY Dr. A . Udhayakumar

Submitted for viva-voce Examination held on ___________________.

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

WE would like to thank all those who gave us their time, care and guidance to which can never be rapid. The following dignitaries deserve a special mention. We express our heartfelt gratitude to our honourable Secretary **Shri. UDHAN KUMAR CHORDIA** and Associate Secretary **Shri. HEMANT P CHORDIA** for their continuous encouragement and providing all the resources needed behind the scenes. We would like to express our sincere thanks to our beloved principal **Dr. N. VENKATARAMANAN**, Vice Principal **Dr. B. MAHAVIR**, Dean **Dr. M . M. RAMYA**, Deputy Dean **Dr. R. SUREKHA**. We wish to thank our Facilitator **Dr. A. UDHAYAKUMAR.,M.Sc.,MCA., M.Phil., SET., NET., Ph.D.,** Department of Computer Applications for his guidance.

Special thanks to our Guide **Dr. A. CATHREEN GRACIAMARY**.,**MCA., M.Phil., SET., NET., Ph.D.,** Department of Computer Applications, Agurchand Manmull Jain College for her unwavering support and involvement in every part of our project.

SANDEEP B NAMBIAR

PRITHIV RAJ K

RITESH J.L

TABLE OF CONTENT

ABSTRACT

The project is titled as PCB which is the abbreviation for Personal Computer Builder. The project is built using python for its backend and logical performance which also holds the coding blocks of the program and uses html for its front end.The main objective of our project is to use the above mentioned tools and create a website which helps the user to build a personal computer by the basis of his/her wish. Any type of customer with internet connection and android or any windows devices can be used to access the website.The predecessor of the idea has published this in a way where the user can buy the products which they like,but the main vision of the project is that to acknowledge people about the avoidance of wastage of money in above the par level or below it.
The timeline of our project has been defined by using the python language which also holds the data about the system specifications for users according to their usage of it. The finalised specification then is displayed via output module and acknowledged to the users. The information has been cross-referenced with authentic pc component websites such as INTEL,AMD,NVIDIA for the declaration of the benchmarks. The website is also working forwards for the assembling of the desired pc and delivering it to the users with no service cost included.

1.INTRODUCTION

The project PCB(Personal Computer Builder) works on the objective of giving information on the motive of a help provider where people will have and be able to get ideas about the hardware specifications used in their computers which enables them in the free flow use of the system. The project has been designed in a way where the users are divided into three major categories such as High-end,Medium-end,Low-end.
This division is possible because of the user input budget where 25,000 is kept as a starting range for building PCs and kept in low end users till 25,000.
Price ranging from 25,000-70,000 has been calculated as medium-end user and price range from 70,000-8,00,000 has been placed under the category of High-end user.
Thus,The website acts as a guide for the user who has no knowledge or idea about computer hardware and its combination.

Flask has been used to interconnect the database which is available in the source code. That is the list of pc components which is planned according to the user.

2.SYSTEM ANALYSIS

2.1.EXISTING SYSTEM

The project has been influenced by various real world problems and websites some of them are

1. Pcstudio.in
2. Computechstore.in
3. bitkart.com

Few of the advantages of the system from its predecessor

- People have the right to choose their wished product with its benchmarks.
- The price of the product will be less compared with other sites as we have our own products bought and well maintained with invoices included with gst.
- All over India free delivery is available.
- 20% off for persons who are going to purchase in bulk orders i.e(more than 7).

2.2.PROPOSED SYSTEM

A project proposal is a project management document that's used to define the objectives and requirements of a project. It helps organisations and external project stakeholders agree on an initial project planning framework.
The main purpose of a project proposal is to get buy-in from decision-makers. That's why a project proposal outlines your project's core value proposition; it sells value to both internal and external project stakeholders. The intent of the proposal is to grab the attention of stakeholders and project sponsors. Then, the next step is getting them excited about the project summary.

The project PCB(Personal Computer Builder) has been proposed to act as an information provider for the user who uses this website for building pc. People with basic or no knowledge in the specifications of personal computer hardware are the ones who need the knowledge about the hardware parts which have been used in their pc. Hence providing them the information/knowledge about the usage and hardware parts.

2.3.FEASIBILITY STUDY

The are key aspects to consider in the feasibility study:

2.3.1 Technical Feasibility:

Assess the technical requirements and constraints of developing and deploying the PC Builder program. Consider factors such as:
Compatibility with different operating systems (Windows, macOS, Linux).
Integration with databases for storing component and user data.
Scalability to handle a large number of users and configurations.
Performance requirements, such as response times for fetching component information and calculating configuration prices.

2.3.2 Market Feasibility:

Analyse the market demand for PC building software and customization tools. Consider factors such as:
Market trends in the PC hardware industry, including the popularity of DIY PC building.
Competitor analysis to identify existing PC builder tools and their features.
Target audience demographics and preferences, such as gamers, enthusiasts, and professionals.

2.3.3 Financial Feasibility:

Evaluate the financial aspects of developing and maintaining the PC Builder program. Consider factors such as:
Development costs, including software development, database setup, and infrastructure.
Operating costs, including hosting, maintenance, and support.
Revenue generation strategies, such as subscription fees, premium features, or affiliate partnerships.
Projected return on investment (ROI) and break-even analysis.

2.3.4 Legal and Compliance Feasibility:

Ensure compliance with legal and regulatory requirements related to software development and operation. Consider factors such as:
Intellectual property rights for software code, databases, and content.
Data privacy and security regulations, such as GDPR or CCPA, for handling user data.
Terms of service and user agreements to protect the rights and responsibilities of users and the software provider.

2.3.5 Operational Feasibility:

Assess the operational aspects of implementing and using the PC Builder program. Consider factors such as:
User interface design and usability testing to ensure ease of use for a diverse range of users.
Training and support requirements for users and administrators.
Availability of resources, such as skilled developers, designers, and support staff.
Risk management strategies to address potential challenges and mitigate project risks.

2.3.6 Environmental and Social Feasibility:

Consider the environmental and social impacts of developing and operating the PC Builder program. Consider factors such as:
Environmental sustainability practices in software development and data centre operations.
Social responsibility initiatives, such as diversity and inclusion in hiring practices and community engagement programs.

3.SYSTEM SPECIFICATIONS

3.1.HARDWARE REQUIREMENTS

Proper Internet connection (WIFI or Broadband)
Laptops,Personal Computer

3.2.SOFTWARE REQUIREMENTS

Any Android and windows devices
Chrome

- Using these requirements any user can log in into our website and get information about their personal space requirements.
- The site won't be accessible while we are undergoing maintenance.
- In the future we have planned to launch this app as an application which requires android version minimum of 8.0(oreo) and windows version of windows 10.

4.SOFTWARE DESIGN

4.1.DATA FLOW DIAGRAM(DFD)

```
+-----------------------+
|  User Interaction    |
+---------+-------------+
          |
          v
+---------+--------------+
| Input Processing      |
+---------+--------------+
          |
          v
 +---------+---------------+
 | Price Calculation      |
 +---------+---------------+
          |
          v
 +---------+-----------------+
 | Database Handling      |
 +---------+-----------------+
          |
          v
 +---------+--------------+
 |   Configuration        |
 +---------+--------------+
          |
          v
 +---------+----------------+
 |Component Display     |
 +---------+----------------+
          |
          v
 +---------+----------+
  | Result Display  |
 +--------------------+
```

Figure 4.1 DFD of PC builder

EXPLANATION:

User Interaction: External entities interact with the PC Builder program through various means such as a web interface or a command-line interface.

Input Processing: Processes user input, including budget, preferences, and component selections.

Database Handling: Manages data stored in the database, including user information, component details, and configurations.

Component Display: Retrieves component information from the database and displays it to the user for selection.

Configuration: Handles the selection of components by the user to create a PC configuration.

Price Calculation: Calculates the total price of the selected components based on their individual prices.

Result Display: Presents the final PC configuration, including selected components and total price, to the user.

4.2.ENTITY RELATIONSHIP DIAGRAM(ER)

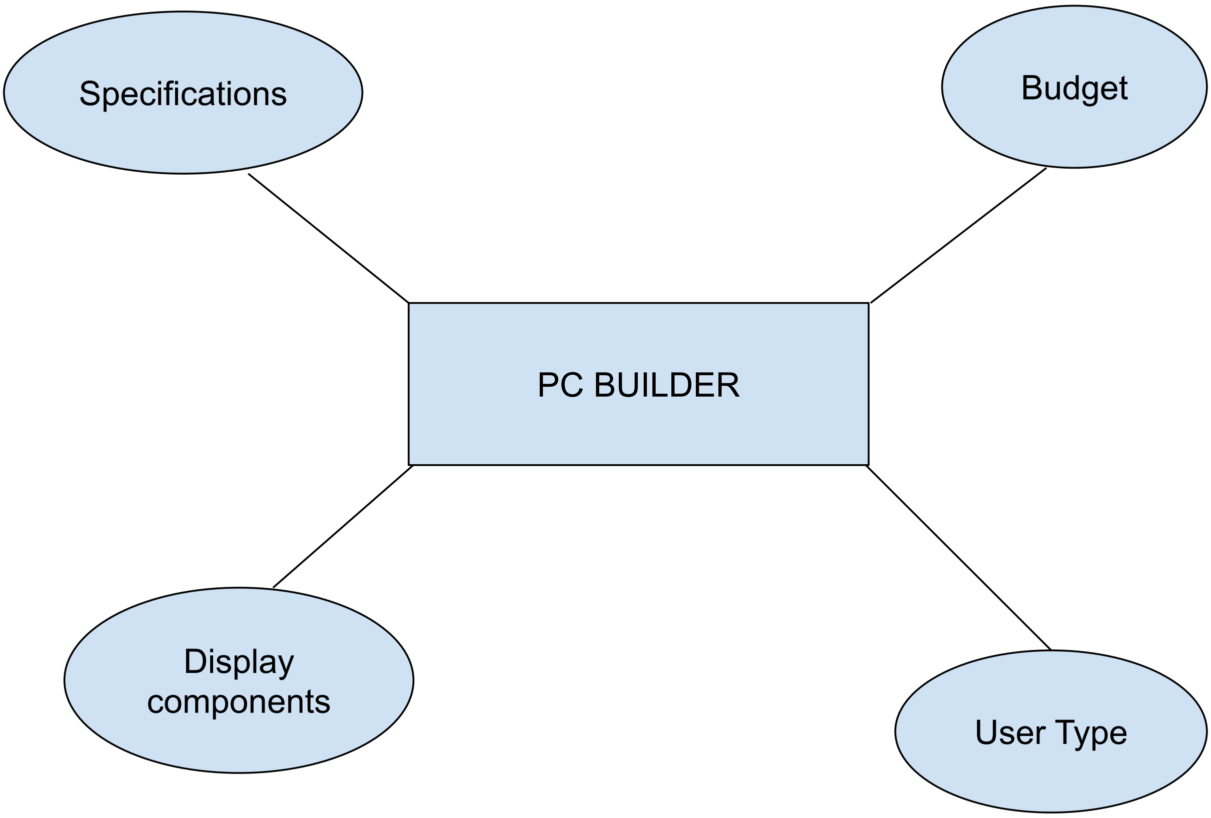


Figure 4.2 ER of PC builder

EXPLANATION:

USERS: Represents the users of the PC Builder program. Each user can create multiple configurations**.**

Attributes: user_id (primary key), username, email, password.
Components: Represents the various components available for building a PC **(e.g., CPU, GPU, RAM, etc.).**

Attributes: component_id (primary key), name, type (e.g., CPU, GPU, RAM), price.
Configurations: Represents the configurations created by users, which consist of selected components.

Attributes: config_id (primary key), user_id (foreign key referencing Users), component_id (foreign key referencing Components), price.
This ER diagram illustrates the relationships between the main entities in the PC Builder program:

Each user can create multiple configurations (one-to-many relationship between Users and Configurations).
Each configuration can include multiple components, and each component can be included in multiple configurations (many-to-many relationship between Components and Configurations, resolved using an associative entity).

Keep in mind that this is a simplified ER diagram, and the actual structure may vary based on the specific requirements and functionalities of your PC Builder program. Additionally, you may have additional entities and relationships depending on the complexity of your application.

5.SOFTWARE DESCRIPTION

The PC Builder Program is a software application designed to assist users in selecting and configuring components to build their own customised personal computers. It provides users with a user-friendly interface to explore various hardware options, select components based on their preferences and budget, and create complete configurations tailored to their specific needs.

5.1 Key Features:

Component Selection: The program offers a comprehensive database of hardware components including CPUs, GPUs, RAM, storage devices, motherboards, power supplies, and more. Users can browse through these components, view detailed specifications, and select the ones that best fit their requirements.

5.2 Configuration Creation:

Users can create and save multiple configurations by selecting their desired components. The program calculates the total cost of the configuration in real-time, helping users stay within their budget constraints.

5.3 Budget Management:

The program includes features to assist users in managing their budgets effectively. It provides tools to compare prices, explore cost-effective alternatives, and adjust component selections to meet budgetary constraints.

5.4 Compatibility Checking:

The program automatically checks for compatibility between selected components to ensure that users build configurations that work seamlessly together. It alerts users of any potential compatibility issues and provides recommendations for resolving them.

5.5 Customization Options:

Users have the flexibility to customise various aspects of their configurations such as overclocking settings, cooling solutions, form factors, and aesthetics. The program provides options to personalise configurations according to individual preferences.

5.6 Data Persistence:

The program stores user profiles and configuration data securely, allowing users to access and modify their configurations at any time. It also offers features for exporting and sharing configurations with others.

5.7 Educational Resources:

The program includes educational resources and guides to help users learn about PC hardware, understand technical specifications, and make informed decisions when selecting components.

5.8 Target Audience:

The PC Builder Program is designed for a diverse audience including PC enthusiasts, gamers, professionals, and DIY enthusiasts who are interested in building their own custom PCs. It caters to users with varying levels of technical expertise, from beginners to advanced users.

5.9 Platform Compatibility:

The PC Builder Program is compatible with multiple platforms including Windows, macOS, and Linux. It is available as a standalone desktop application that can be installed locally on the user's computer.

.

6.DATABASE DESIGN

Table 6.1 data used in the program

User Type	Motherboard	Ram	SSD/HDD	Intel	Graphics card	AMD
Low-End user	ASUS ROG Strix Z590	32GB or 64GB DDR4 3400MHz - 4500MHz	Samsung 970 EVO Plus 1TB/ Seagate Barracuda 2TB	Intel i5-11600K	NVIDIA GTX 1650 or AMD RX 5500 XT	AMD Ryzen 9 5950X
Medium-End user	MSI B450 TOMAHAWK MAX	16GB DDR4 3400MHz	Kingston KC2500 1TB/ Seagate Barracuda 1TB	Intel i7-12700K	NVIDIA RTX 3060 or AMD RX 6700 XT	AMD Ryzen 7 5800X
High-End user	ASRock B450M-HDV R4.0	8GB DDR4 2400MHz - 3400MHz	Kingston A2000 250GB/WD Blue 500GB	Intel i9-11900K	NVIDIA RTX 3070 or AMD RX 6800	AMD Ryzen 5 5600X

We might have come across the word **"Database"** quite often. This term carries a high emphasis on its arms. More often, it is not just related to the developer's perspective but is quite often used with non-tech groups or communities. Technically, a database is more of a storage term used to denote the relationship with different forms of data that are coagulated in a single place. Thus, we can define a database as an organised collection of data, generally stored and accessed electronically through computer systems This article is highly centric to the database design and its association with citable terms and methodologies was commonly taken into account. We'll be discussing those terms concerning database design to understand the bits and pieces. Let's talk about it straight away.

Database design can be generally defined as a collection of tasks or processes that enhance the designing, development, implementation, and maintenance of enterprise data management systems. Designing a proper database reduces the maintenance cost thereby improving data consistency and the cost-effective measures are greatly influenced in terms of disk storage space. Therefore, there has to be a brilliant concept of designing a database. The designer should follow the constraints and decide how the elements correlate and what kind of data must be stored.

The main objectives behind database designing are to produce physical and logical design models of the proposed database system. To elaborate this, the logical model is primarily concentrated on the requirements of data and the considerations must be made in terms of monolithic considerations and hence the stored physical data must be stored independent of the physical conditions.

On the other hand, the physical database design model includes a translation of the logical design model of the database by keeping control of physical media using hardware resources and software systems such as Database Management System (DBMS).

7.PROJECT DESCRIPTION

7.1.MODULE LIST:

In the context of a software application like a PC Builder program, "module details" typically refer to the different modules or components of the software system. These modules encapsulate specific functionality and may interact with each other to achieve the overall objectives of the program. Below are some typical modules that may be present in a PC Builder program along with their functionalities:

- User Interface Module
- Database Module
- Component Management Module
- Configuration Module
- Educational Resources Module

7.2.MODULE DESCRIPTION:

7.2.1.User Interface Module:

Responsible for presenting the user interface to the user.
Handles user input and interactions.
Displays information about available components and configurations.
Provides forms for users to input their preferences and budget.

7.2.2.Database Module:

Manages the storage and retrieval of data related to users, components, and configurations.
Handles database operations such as querying, inserting, updating, and deleting records.
Ensures data integrity and security.

7.2.3.Component Management Module:

Handles the management of hardware components available for selection.
Retrieves component data from the database.
Provides functions for filtering and sorting components based on various criteria.
Manages component compatibility checks.

7.2.4.Configuration Module:

Facilitates the creation, modification, and deletion of PC configurations.
Calculates the total price of configurations based on selected components.
Performs validation to ensure configurations meet user requirements and budget constraints.

7.2.5.Educational Resources Module:

Offers educational content such as articles, guides, and tutorials on PC hardware.
Provides tips, recommendations, and best practices for building custom PCs.
Helps users make informed decisions when selecting components.

8.SYSTEM IMPLEMENTATION

8.1.SAMPLE CODING

PYTHON CODE:

```
class PCBuilder:
    def __init__(self):
        self.specifications = {}
        self.intel_processors = ["Intel i9-11900K", "Intel i7-12700K", "Intel i5-11600K"]
        self.amd_processors = ["AMD Ryzen 9 5950X", "AMD Ryzen 7 5800X", "AMD
Ryzen 5 5600X"]
        self.dollar_to_rupee_conversion_rate = 75.0  # You can adjust the conversion rate
as needed
        self.graphics_card_recommendations = {
            (0, 25000): "Integrated Graphics (e.g., Intel UHD Graphics, AMD Radeon
Graphics)",
            (25001, 50000): "NVIDIA GTX 1650 or AMD RX 5500 XT",
            (50001, 100000): "NVIDIA RTX 3060 or AMD RX 6700 XT",
            (100001, 150000): "NVIDIA RTX 3070 or AMD RX 6800",
            (150001, 200000): "NVIDIA RTX 3080 or AMD RX 6900 XT",
            (200001, float('inf')): "NVIDIA RTX 3090 or AMD RX 6900 XT (for enthusiasts)"
        }

    def get_user_requirements(self):
        print("Welcome to the PC Builder!")
        print("Please answer the following questions to help us understand your
requirements.")

        self.specifications['budget'] = float(input("1. What is your budget? ₹"))
        self.specifications['usage'] = input("2. What will you use the PC for? (e.g., gaming,
video editing, programming): ").lower()
        self.specifications['storage'] = input("3. Do you need a large amount of storage?
(yes/no): ").lower()

        if self.specifications['storage'] == 'yes':
            self.specifications['storage_type'] = input("4. Do you prefer SSD or HDD for
storage? (ssd/hdd): ").lower()
```

```
        self.specifications['graphics_card'] = input("5. Do you need a dedicated graphics
card? (yes/no): ").lower()
        self.specifications['processor_preference'] = input("6. Do you have a preference
for Intel or AMD processors? (intel/amd/none): ").lower()

    def categorize_user(self):
        if 0 <= self.specifications['budget'] <= 25000:
            return "Low-End User"
        elif 40000 <= self.specifications['budget'] <= 75000:
            return "Medium-End User"
        elif 75000 <= self.specifications['budget'] <= 600000:
            return "High-End User"
        else:
            return "Undefined User"

    def get_graphics_card_recommendation(self, budget):
        for price_range, recommendation in self.graphics_card_recommendations.items():
            if price_range[0] <= budget <= price_range[1]:
                return recommendation

    def build_pc(self):
        print("\nBased on your requirements, here's a suggested PC configuration:")

        print(f"\nBudget: ₹{self.specifications['budget']:.2f}")
        print(f"User Category: {self.categorize_user()}")

        user_category = self.categorize_user()

        if user_category != "Undefined User":
            if self.specifications['usage'] == 'gaming':
                print("For gaming, we recommend a powerful CPU and a dedicated graphics
card.")
                graphics_card_recommendation =
self.get_graphics_card_recommendation(self.specifications['budget'])
                print(f"Graphics Card Recommendation: {graphics_card_recommendation}")

                # Sample motherboard and RAM suggestions (customize based on actual
components)
                if user_category == "High-End User":
                    print("Motherboard: High-end gaming motherboard (e.g., ASUS ROG Strix
Z590)")
                    print("RAM: 32GB or 64GB DDR4 3400MHz - 4500MHz")

                elif user_category == "Medium-End User":
```

```
                print("Motherboard: Mid-range gaming motherboard (e.g., MSI B450
TOMAHAWK MAX)")
                print("RAM: 16GB DDR4 3400MHz")

            elif user_category == "Low-End User":
                print("Motherboard: Entry-level gaming motherboard (e.g., ASRock
B450M-HDV R4.0)")
                print("RAM: 8GB DDR4 2400MHz - 3400MHz")

        elif self.specifications['usage'] == 'video editing':
            print("For video editing, consider a powerful CPU and additional RAM for
smooth performance.")
            graphics_card_recommendation =
self.get_graphics_card_recommendation(self.specifications['budget'])
            print(f"Graphics Card Recommendation: {graphics_card_recommendation}")

            # Sample motherboard and RAM suggestions (customize based on actual
components)
            if user_category == "High-End User":
                print("Motherboard: High-end video editing motherboard (e.g., ASUS
ProArt B550-CREATOR)")
                print("RAM: 32GB or 64GB DDR4 3400MHz - 4500MHz")

            elif user_category == "Medium-End User":
                print("Motherboard: Mid-range video editing motherboard (e.g., MSI MAG
B550 TOMAHAWK)")
                print("RAM: 16GB DDR4 3400MHz")

            elif user_category == "Low-End User":
                print("Motherboard: Entry-level video editing motherboard (e.g., ASRock
B450M Pro4)")
                print("RAM: 8GB DDR4 2400MHz - 3400MHz")

        elif self.specifications['usage'] == 'programming':
            print("For programming, a balanced configuration with a decent CPU and
sufficient RAM is recommended.")

            # Sample motherboard and RAM suggestions (customize based on actual
components)
            if user_category == "High-End User" or user_category == "Medium-End
User":
                print("Motherboard: Mid-range programming motherboard (e.g., ASUS
PRIME B450M-A)")
                print("RAM: 16GB DDR4 3400MHz")
            elif user_category == "Low-End User":
                print("Motherboard: Entry-level programming motherboard (e.g., ASRock
B450M-HDV R4.0)")
                print("RAM: 8GB DDR4 2400MHz - 3400MHz")

        else:
            print(f"We recommend a general-purpose configuration for
{self.specifications['usage']}.")

        if self.specifications['storage'] == 'yes':
            print(f"Storage Type: {self.specifications['storage_type'].upper()}")

            if self.specifications['storage_type'] == 'ssd':
                print("SSD Options:")

                # Sample SSD options based on budget (customize based on actual
components)
                if user_category == "High-End User":
                    print("1. Samsung 970 EVO Plus 1TB")
                    print("2. WD Black SN850 1TB")
                    print("3. Kingston KC2500 1TB")

                elif user_category == "Medium-End User":
                    print("1. Crucial P5 500GB")
                    print("2. Kingston A2000 500GB")
                    print("3. WD Blue SN550 500GB")

                elif user_category == "Low-End User":
                    print("1. Kingston A2000 250GB")
                    print("2. WD Green 240GB")
                    print("3. Crucial BX500 240GB")

            elif self.specifications['storage_type'] == 'hdd':
                print("HDD Options:")

                # Sample HDD options based on budget (customize based on actual
components)
                if user_category == "High-End User":
                    print("1. Seagate Barracuda 2TB")
                    print("2. WD Black 2TB")
                    print("3. Toshiba X300 2TB")

                elif user_category == "Medium-End User":
                    print("1. Seagate Barracuda 1TB")
                    print("2. WD Blue 1TB")
                    print("3. Toshiba P300 1TB")

                elif user_category == "Low-End User":
                    print("1. Seagate Barracuda 500GB")
                    print("2. WD Blue 500GB")
                    print("3. Toshiba P300 500GB")

        if self.specifications['graphics_card'] == 'yes':
            print("A dedicated graphics card will enhance your system's graphics
performance.")

        if self.specifications['processor_preference'] == 'intel':
            print("Consider an Intel processor for optimal performance.")
            print("Available Intel processors:")
            for processor in self.intel_processors:
                print(f"- {processor}")

            # Sample motherboard suggestions for Intel (customize based on actual
components)
            if user_category == "High-End User":
                print("Motherboard Options:")
                print("1. ASUS ROG Strix Z590")
                print("2. MSI MPG B560 GAMING EDGE WIFI")

            elif user_category == "Medium-End User":
                print("Motherboard Options:")
                print("1. MSI MAG B450 TOMAHAWK MAX")
                print("2. ASUS PRIME B460M-A")

            elif user_category == "Low-End User":
                print("Motherboard Options:")
                print("1. ASRock B450M-HDV R4.0")
                print("2. ASUS Prime H410M-E")

        elif self.specifications['processor_preference'] == 'amd':
            print("Consider an AMD processor for a cost-effective and powerful option.")
            print("Available AMD processors:")
            for processor in self.amd_processors:
                print(f"- {processor}")

            # Sample motherboard suggestions for AMD (customize based on actual
components)
```

```
                if user_category == "High-End User":
                    print("Motherboard Options:")
                    print("1. ASUS ProArt B550-CREATOR")
                    print("2. MSI MPG B550 GAMING PLUS")

                elif user_category == "Medium-End User":
                    print("Motherboard Options:")
                    print("1. MSI MAG B550 TOMAHAWK")
                    print("2. ASUS TUF B450M-PLUS GAMING")

                elif user_category == "Low-End User":
                    print("Motherboard Options:")
                    print("1. ASRock B450M Pro4")
                    print("2. Gigabyte B450M DS3H")

                print("We'll provide recommendations based on your other requirements.")

            # Caution message for budget mismatch
            if (user_category == "Low-End User" and not 0 <= self.specifications['budget']
<= 25000) or \
               (user_category == "Medium-End User" and not 40000 <=
self.specifications['budget'] <= 75000) or \
               (user_category == "High-End User" and not 75000 <=
self.specifications['budget'] <= 600000):
                print("\nCaution: It seems that the specified budget may not align with the
selected usage or preferences.")
                print("You might be paying more for certain specifications than necessary.")

        else:
            print("Undefined User: Please provide a valid budget within the specified
ranges.")

        print("\nThank you for using the PC Builder!")

if __name__ == "__main__":
    pc_builder = PCBuilder()
    pc_builder.get_user_requirements()
    pc_builder.build_pc()
```

FLASK:

```
# app.py

from flask import Flask, render_template, request
from pcbuilder2 import PCBuilder  # Import your PC Builder program

app = Flask(__name__)
pc_builder = PCBuilder()  # Initialize PCBuilder instance


@app.route('/')
def index():
   return render_template('index.html')

@app.route('/build_pc', methods=['POST'])
def build_pc():
   # Retrieve user inputs from the form
   budget = float(request.form['budget'])
   usage = request.form['usage']
   storage = request.form['storage']
   storage_type = request.form['storage_type']
   graphics_card = request.form['graphics_card']
   processor_preference = request.form['processor_preference']

   # Set user specifications
   pc_builder.specifications['budget'] = budget
   pc_builder.specifications['usage'] = usage
   pc_builder.specifications['storage'] = storage
   pc_builder.specifications['storage_type'] = storage_type
   pc_builder.specifications['graphics_card'] = graphics_card
   pc_builder.specifications['processor_preference'] = processor_preference

   # Build PC and get recommendations
   recommendation = pc_builder.build_pc()
   return render_template('result.html',recommendations=recommendation)


if __name__ == '__main__':
   app.run(debug=True)
```

HTML:

INDEX.html:-

```
<!-- index.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PC Builder</title>
</head>
<body>
    <h1>Welcome to PC Builder</h1>
    <form action="/build_pc" method="post">
        <label for="budget">Budget (₹):</label>
        <input type="number" id="budget" name="budget" required><br><br>

        <label for="usage">Usage:</label>
        <input type="text" id="usage" name="usage" required><br><br>

        <label for="storage">Large Storage Needed?</label>
        <select id="storage" name="storage" required>
            <option value="yes">Yes</option>
            <option value="no">No</option>
        </select><br><br>

        <label for="storage_type">Storage Type:</label>
        <select id="storage_type" name="storage_type" required>
            <option value="ssd">SSD</option>
            <option value="hdd">HDD</option>
        </select><br><br>

        <label for="graphics_card">Dedicated Graphics Card Needed?</label>
        <select id="graphics_card" name="graphics_card" required>
            <option value="yes">Yes</option>
            <option value="no">No</option>
        </select><br><br>

        <label for="processor_preference">Processor Preference:</label>
        <select id="processor_preference" name="processor_preference" required>
            <option value="intel">Intel</option>
            <option value="amd">AMD</option>
        </select><br><br>

        <button type="submit">Build PC</button>
    </form>
</body>
</html>
```

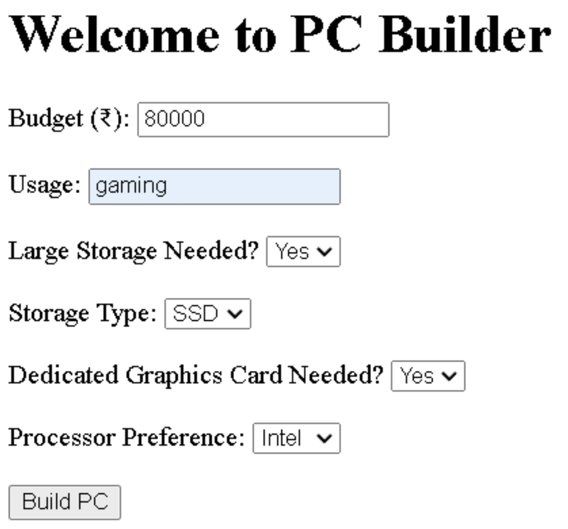
RESULT.html:-

```
<!-- result.html -->

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PC Builder - Results</title>
</head>
<body>
    <h1>PC Builder Results</h1>
    <ul>
        {% for recommendation in recommendations %}
            <li>{{ recommendation }}</li>
        {% endfor %}
    </ul>
</body>
</html>
```

8.2.SCREEN LAYOUT:

A)Getting input from user(HIGH END USER)



Welcome to PC Builder

Budget (₹): 80000

Usage: gaming

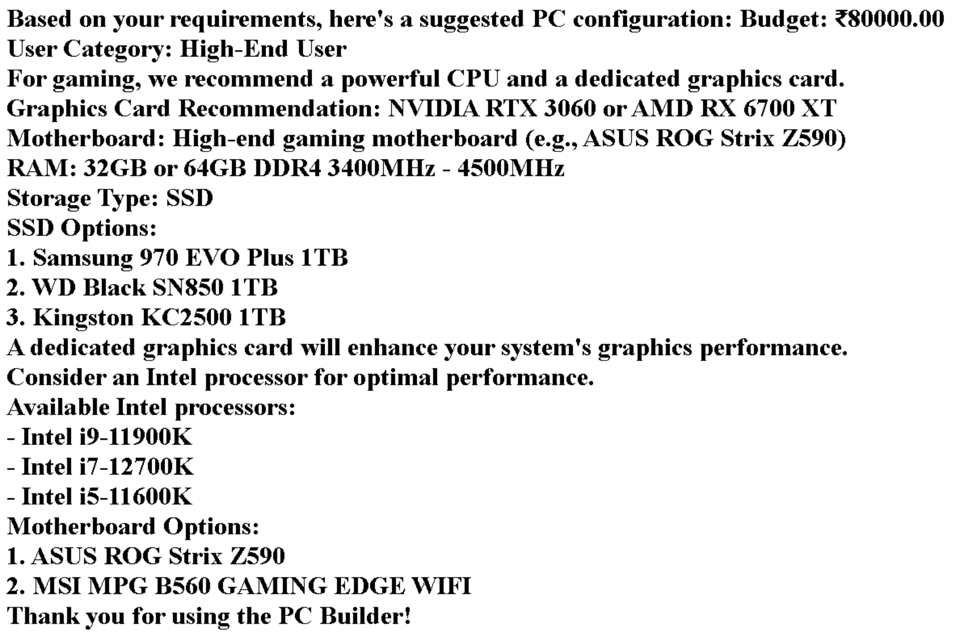
Large Storage Needed? Yes

Storage Type: SSD

Dedicated Graphics Card Needed? Yes

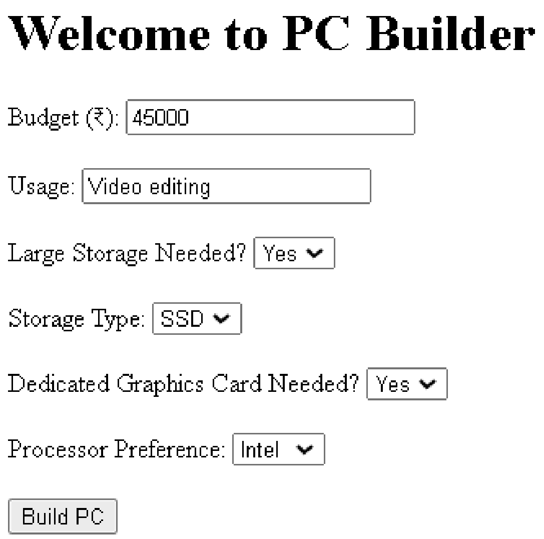
Processor Preference: Intel

B) Displaying output to user



Based on your requirements, here's a suggested PC configuration: Budget: ₹80000.00
User Category: High-End User
For gaming, we recommend a powerful CPU and a dedicated graphics card.
Graphics Card Recommendation: NVIDIA RTX 3060 or AMD RX 6700 XT
Motherboard: High-end gaming motherboard (e.g., ASUS ROG Strix Z590)
RAM: 32GB or 64GB DDR4 3400MHz - 4500MHz
Storage Type: SSD
SSD Options:
1. Samsung 970 EVO Plus 1TB
2. WD Black SN850 1TB
3. Kingston KC2500 1TB
A dedicated graphics card will enhance your system's graphics performance.
Consider an Intel processor for optimal performance.
Available Intel processors:
- Intel i9-11900K
- Intel i7-12700K
- Intel i5-11600K
Motherboard Options:
1. ASUS ROG Strix Z590
2. MSI MPG B560 GAMING EDGE WIFI
Thank you for using the PC Builder!

C) Getting input from user(MEDIUM END)



Welcome to PC Builder

Budget (₹): 45000

Usage: Video editing

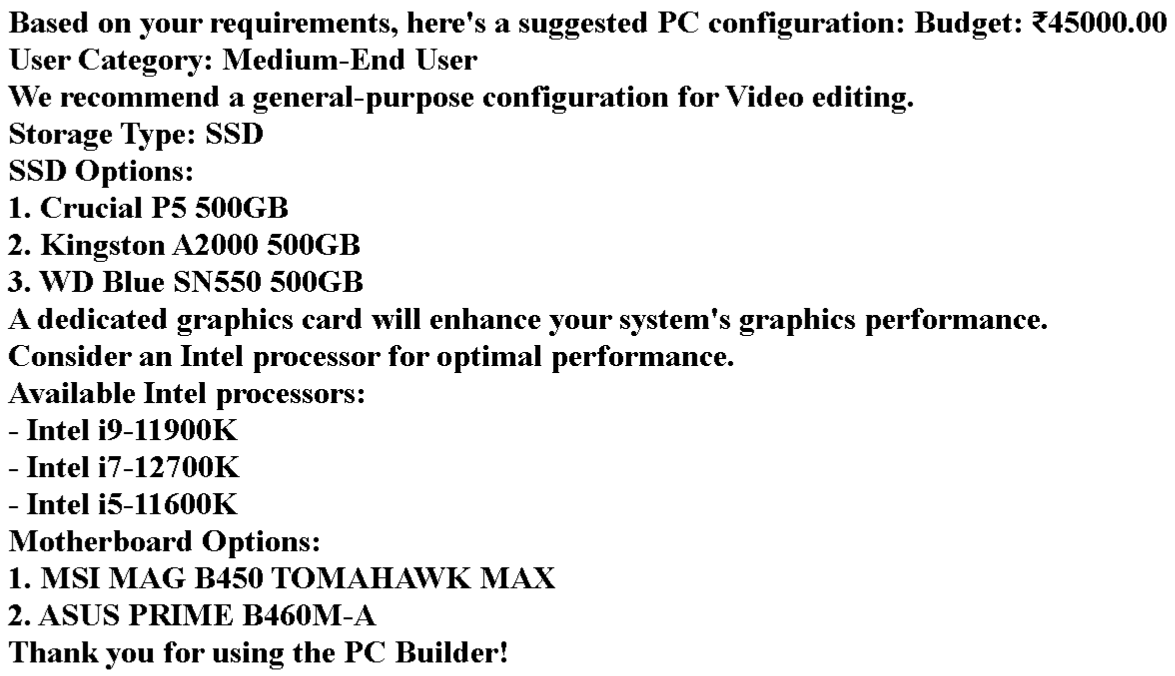
Large Storage Needed? Yes

Storage Type: SSD

Dedicated Graphics Card Needed? Yes

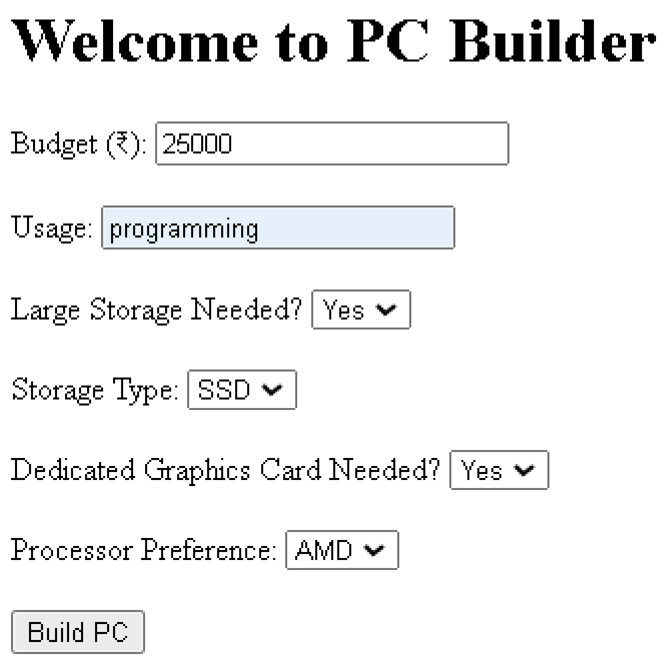
Processor Preference: Intel

D) Displaying output to user



```
Based on your requirements, here's a suggested PC configuration: Budget: ₹45000.00
User Category: Medium-End User
We recommend a general-purpose configuration for Video editing.
Storage Type: SSD
SSD Options:
1. Crucial P5 500GB
2. Kingston A2000 500GB
3. WD Blue SN550 500GB
A dedicated graphics card will enhance your system's graphics performance.
Consider an Intel processor for optimal performance.
Available Intel processors:
- Intel i9-11900K
- Intel i7-12700K
- Intel i5-11600K
Motherboard Options:
1. MSI MAG B450 TOMAHAWK MAX
2. ASUS PRIME B460M-A
Thank you for using the PC Builder!
```

E)Getting input from user(LOW END)



Welcome to PC Builder

Budget (₹): 25000

Usage: programming

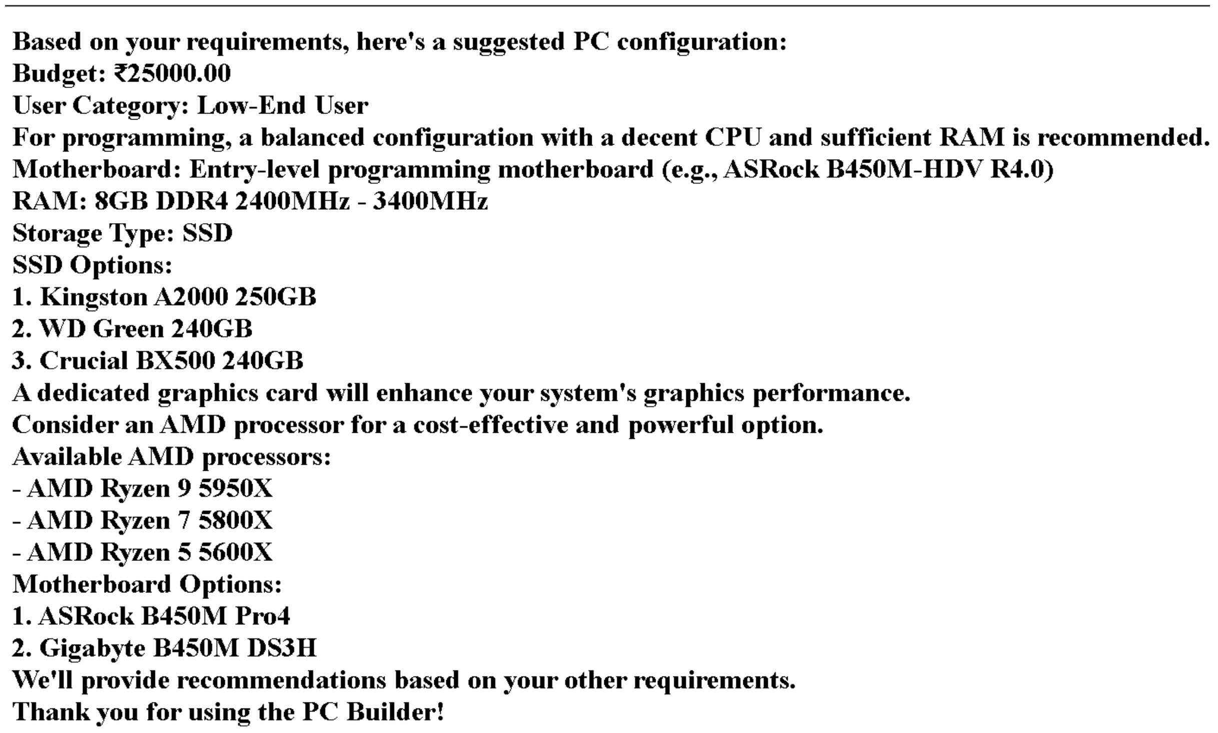
Large Storage Needed? Yes

Storage Type: SSD

Dedicated Graphics Card Needed? Yes

Processor Preference: AMD

F) Displaying output to user



```
Based on your requirements, here's a suggested PC configuration:
Budget: ₹25000.00
User Category: Low-End User
For programming, a balanced configuration with a decent CPU and sufficient RAM is recommended.
Motherboard: Entry-level programming motherboard (e.g., ASRock B450M-HDV R4.0)
RAM: 8GB DDR4 2400MHz - 3400MHz
Storage Type: SSD
SSD Options:
1. Kingston A2000 250GB
2. WD Green 240GB
3. Crucial BX500 240GB
A dedicated graphics card will enhance your system's graphics performance.
Consider an AMD processor for a cost-effective and powerful option.
Available AMD processors:
- AMD Ryzen 9 5950X
- AMD Ryzen 7 5800X
- AMD Ryzen 5 5600X
Motherboard Options:
1. ASRock B450M Pro4
2. Gigabyte B450M DS3H
We'll provide recommendations based on your other requirements.
Thank you for using the PC Builder!
```

8.3.REPORT

The project PCB has thus successfully been executed with the clause of RS 80000 as a budget for building a gaming personal computer with the best available specifications,available in the market with the user defined specifications who has been given it through the front page of the site.
Then the website is redirected to the result content of the website and displays the information.Thus, fulfilling the user wished personal computer**.**

Few of the expected scenarios for the three available cases is High end usage output,Medium end usage output and Low end usage output. All of the cases output has been published successfully in the above scenarios. The case of medium end users the system divides the price range after the details of budget is 45000 and the low end user is divided into the budget 25000 which gives the output according to its budgets in the above given outputs. Thus the reporting can be concluded after witnessing the output of the three cases.

9.SYSTEM TESTING

9.1.UNIT TESTING:

Unit testing is a type of software testing that focuses on individual units or components of a software system. The purpose of unit testing is to validate that each unit of the software works as intended and meets the requirements. Unit testing is typically performed by developers, and it is performed early in the development process before the code is integrated and tested as a whole system.

Unit tests are automated and are run each time the code is changed to ensure that new code does not break existing functionality. Unit tests are designed to validate the smallest possible unit of code, such as a function or a method, and test it in isolation from the rest of the system. This allows developers to quickly identify and fix any issues early in the development process, improving the overall quality of the software and reducing the time required for later testing.

9.2.SYSTEM TESTING:

System Testing is a type of software testing that is performed on a complete integrated system to evaluate the compliance of the system with the corresponding requirements. In system testing, integration testing passed components are taken as input. The goal of integration testing is to detect any irregularity between the units that are integrated together. System testing detects defects within both the integrated units and the whole system.

The result of system testing is the observed behaviour of a component or a system when it is tested. System Testing is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. System testing tests the design and behaviour of the system and also the expectations of the customer. It is performed to test the system beyond the bounds mentioned in the software requirements specification (SRS).

System Testing is basically performed by a testing team that is independent of the development team that helps to test the quality of the system impartially. It has both functional and non-functional testing. System Testing is a black-box testing.

9.3.INTEGRATION TESTING:

Integration testing is the process of testing the interface between two software units or modules. It focuses on determining the correctness of the interface. The purpose of integration testing is to expose faults in the interaction between integrated units. Once all the modules have been unit-tested, integration testing is performed.

Integration testing is a software testing technique that focuses on verifying the interactions and data exchange between different components or modules of a software application. The goal of integration testing is to identify any problems or bugs that arise when different components are combined and interact with each other. Integration testing is typically performed after unit testing and before system testing. It helps to identify and resolve integration issues early in the development cycle, reducing the risk of more severe and costly problems later on.

10.CONCLUSION & FUTURE ENHANCEMENT

10.1 CONCLUSION

The project Personal Computer builder makes any user to build their desired computer using the specifications according to their wish and budget. The main idea of this project is to let everyone know about the technology which is available for them according to their wish. Any user who has access to internet can access PCB website

10.2 FUTURE ENHANCEMENT

- In future users can fix an appointment to build their pc.
- Found in English,Tamil,Malayalam,Telugu,Kannada,Hindi.
- Regular users can get a membership in our studio.
- Monthly once users can participate in giveaways,

11.REFERENCES

1. Website for source of information about specs taken from https://thenextweb.com/news/build-desktop-pc-reference-guide

2. Youtube channel https://youtu.be/LEOpT6DRTSI?si=jOG_3sLgycIkOSjK

3.Source code with modification https://chat.openai.com/share/037996da-c808-4112-b9cc-7e6b57ab8a57