**Basavarajeswari group of institutions**

# Ballari Institute of Technology & Management

**AUTONOMOUS INSTITUTE UNDER VISVESVARAYA TECHNOLOGICAL UNIVERSITY JNANASANGAMA,**

**BELAGAVI 590018**

**INTERNSHIP**

**Report On**

**FITNESS CLASS BOOKING SYSTEM**

Submitted in partial fulfillment of the requirements for the award of degree of

# Bachelor of Engineering In

## COMPUTER SCIENCE AND ENGINEERING

### Submitted by

### SHALU KUMARI

**3BR22CS153**

### Internship Carried Out

### By

**EZ TRAININGS & TECHNOLOGIES PVT.LTD**

**HYDERABAD**

**Internal Guide External Guide**

**Mrs. MADHURI A Mr. BALAJI SRINIVASAN**

**Assistant Professor ,CSE Sr. Faculty**

**Ms. SAMEENA YASME**

**Supervisor ,CSE**

#### BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

NACC Accredited Institution\*

**(RecognizedbyGovt.ofKarnataka,approvedbyAICTE,NewDelhi&AffiliatedtoVisvesvarayaTechnologicalUniversity,Belagavi)**

**"JnanaGangotri"Campus,No.873/2,Ballari-HospetRoad,Allipur,Ballar1-583104(Karnataka)(India)Ph:08392–237100/237190,Fax:08392–237197**

### 2023-2024

**Basavarajeswari group of institutions**

## BALLARI INSTITUTE OF TECHNOLOGY & MANAGEMENT

**Autonomous institute under VISVESVARAYA TECHNOLOGICAL UNIVERSITY JNANASANGAMA,**

**BELAGAVI 590018**

NACC Accredited Institution\*

**(RecognizedbyGovt.ofKarnataka,approvedbyAICTE,NewDelhi&AffiliatedtoVisvesvaraya Technological University, Belagavi)**

**"JnanaGangotri"Campus,No.873/2,Ballari-HospetRoad,Allipur,**

**Ballar1-583104(Karnataka)(India) Ph:08392–237100/237190,Fax:08392–237197**

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

**CERTIFICATE**

This is to certify that the Internship entitled **“FITNESS CLASS BOOKING SYSTEM”** has been successfully completed by **SHALU KUMARI** bearing USN **3BR22CSC153** a bonafide student of Ballari Institute of Technology and Management, Ballari. For the partial fulfillment of the requirements for the **Bachelor’s Degree in Computer Science and Engineering** of the VISVESVARAYA TECHNOLOGICAL UNIVERSITY, Belagavi during the academic year 2023-2024.

#### Signature of Internship

#### Co-ordinators

**Signature of HOD**

**Mrs. MADHURI A Dr. R N KULKARNI**

**Assistant Professor ,CSE Professor & HOD,CSE**

&

**Ms. SAMEENA YASMEEN**

**Supervisor ,CSE**

**DECLARATION**

I, **SHALU KUMARI,** second year student of Computer Science and Engineering, Ballari Institute of Technology, Ballari, declare that Internship entitled **FITNESS CLASS BOOKING SYSTEM** is a part of Internship Training successfully carried out by **EZ TECHNOLOGIES & TRAININGS PVT.LTD, Hyderabad** at “**BITM, BALLARI”.** This report is submitted in partial fulfillment of the requirements for the award of the degree, Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belagavi.

**Date : Signature of the Student**

**Place :**

**ACKNOWLEDGEMENT**

The satisfactions that a company the successful completion of my internship on “FITNESS CLASS BOOKING SYSTEM” would be incomplete without the mention of people who made it possible, whose noble gesture, affection, guidance ,encouragement and support crowned my efforts with success. It is my privilege to express my gratitude and respect to all those who inspired me in the completion of my internship.

I am grateful to my respective coordinators “**Mrs.Madhuri A(Asst.prof,CSE) and Ms.Sameena Yasmeen (Supervisor,CSE)**” for their noble gesture ,support co-ordination and valuable suggestions given to me in the completion of Internship.

I also thank **Dr. R N Kulkarni,** HOD , Department of **Computer Science and Engineering** for extending all his valuable support and encouragement.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Chapter Name** | **Page No.** |
| **1** | **Company Profile** | **1** |
| **2** | **Day to day activity(student diary extract)** | **2** |
| **3** | **Abstract** | **3** |
| **4** | **Introduction of the project** | **4** |
| **5** | **Description** | **5-8** |
| **6** | **Algorithm** | **9** |
| **7** | **Output** | **10** |
| **8** | **Conclusion** | **11** |
| **9** | **References** | **12** |

**COMPANY PROFILE**

**Company Name : EZ Trainings and Technologies Pvt. Ltd**.

**Introduction:**

EZ Trainings and Technologies Pvt. Ltd. is a dynamic and innovative organization dedicated to providing comprehensive training solutions and expert development services. Established with a vision to bridge the gap between academic learning and industry requirements, we specialize in college trainings for students, focusing on preparing them for successful placements. Additionally, we excel in undertaking development projects, leveraging cutting-edge technologies to bring ideas to life.

**Mission:**

Our mission is to empower the next generation of professionals by imparting relevant skills and knowledge through specialized training programs. We strive to be a catalyst in the career growth of students and contribute to the technological advancement of businesses through our development projects.

**Services:**

**College Trainings:**

• Tailored training programs designed to enhance the employability of students

• Industry-aligned curriculum covering technical and soft skills.

• Placement assistance and career guidance.

**Development Projects:**

• End-to-end development services, from ideation to execution.

• Expertise in diverse technologies and frameworks.

• Custom solutions to meet specific business needs

**Locations:**

Hyderabad | Delhi NCR At EZ Trainings and Technologies Pvt. Ltd., we believe in transforming potential into excellence

1

**ABSTRACT**

Fitness class booking system consists of two main classes:

* FitnessClass
* FitnessClassManager.

**FitnessClass:**

Represents an individual fitness class with attributes like class ID, class name, maximum capacity, and a dictionary to store bookings.

Provides methods to add bookings (add\_booking), cancel bookings (cancel\_booking), and retrieve bookings (get\_bookings).

**FitnessClassManager:**

Manages multiple fitness classes by storing them in a list.

Offers (CRUD Operations) functionality to add a new fitness class (add\_class), delete an existing class (delete\_class), update class details such as name and capacity (update\_class), and retrieve a list of all class names (display\_classes).

Allows managing bookings for fitness classes (manage\_fitness\_class\_bookings) by booking or canceling slots for users.

Tracks class attendance (track\_class\_attendance) by counting the number of bookings for each class.

Provides a method to retrieve booking details for a specific class (get\_class\_details).

The code includes a set of unit tests (TestFitnessClassBookingSystem) using the unittest framework to ensure the correctness of the implemented functionalities. These tests cover adding, deleting, and updating classes, managing bookings, tracking attendance, and retrieving class details.

3

**INTRODUCTION OF THE PROJECT**

In this Project:

* Fitness classes are represented as objects with attributes like class ID, name, and maximum capacity.
* The FitnessClassManager class manages multiple fitness classes, providing functionalities to add, delete, and update class details.
* Users can book or cancel bookings for classes, with the system tracking attendance by counting the number of bookings.
* The code includes unit tests to ensure the correctness of system functionalities, covering scenarios such as adding/deleting classes, booking/canceling bookings, tracking attendance, and displaying class information.
* Overall, this system provides a robust solution for managing fitness class schedules and bookings, facilitating effective administration and organization within a fitness facility.

4

**MODULE DISCRIPTION**

**FitnessClass Class**

* **\_init\_(self, class\_id, class\_name, max\_capacity)**
* Purpose: Initializes a FitnessClass object with the provided parameters.
* Parameters:

class\_id: Unique identifier for the fitness class.

class\_name: Name of the fitness class.

max\_capacity: Maximum capacity of the fitness class.

* **add\_booking(self, user\_id)**
* Purpose: Adds a booking for a user to the fitness class if there's available space.
* Parameters:

user\_id: Identifier for the user booking the class.

* Returns: True if the booking was successful, raises ValueError otherwise.
* **cancel\_booking(self, user\_id)**
* Purpose: Cancels the booking for a user from the fitness class.
* Parameters:

user\_id: Identifier for the user whose booking is to be canceled.

* Returns: True if the booking was successfully canceled, raises ValueError otherwise.
* **get\_bookings(self)**
* Purpose: Returns the dictionary of bookings for the fitness class.
* Returns: Dictionary containing user bookings.

**FitnessClassManager Class**

* **\_init\_(self)**
* Purpose: Initializes a FitnessClassManager object with an empty list of classes.
* **add\_class(self, fitness\_class)**
* Purpose: Adds a fitness class to the manager's list of classes.
* Parameters:

5

fitness\_class: FitnessClass object to be added.

* Returns: True if the class was added successfully.
* **delete\_class(self, class\_id)**
* Purpose: Deletes a fitness class from the manager's list of classes.
* Parameters:

class\_id: Identifier of the class to be deleted.

* Returns: True if the class was deleted successfully, raises ValueError otherwise.
* **update\_class(self, class\_id, new\_class\_name, new\_max\_capacity)**
* Purpose: Updates the details of a fitness class.
* Parameters:

class\_id: Identifier of the class to be updated.

new\_class\_name: New name for the class.

new\_max\_capacity: New maximum capacity for the class.

* Returns: True if the class was updated successfully, raises ValueError otherwise.
* **manage\_fitness\_class\_bookings(self, class\_id, user\_id, action)**
* Purpose: Manages bookings for a fitness class (booking or canceling).
* Parameters:

class\_id: Identifier of the class to manage bookings for.

user\_id: Identifier of the user performing the action.

* action: Action to perform ("book" or "cancel").
* Returns: True if the action was successful, raises ValueError otherwise.
* **track\_class\_attendance(self)**
* Purpose: Tracks the attendance for all fitness classes managed by the manager.
* Returns: Dictionary containing class IDs as keys and the number of bookings as values.
* **display\_classes(self)**
* Purpose: Displays the names of all fitness classes managed by the manager.
* Returns: List of class names.

6

* **get\_class\_details(self, class\_id)**
* Purpose: Retrieves details of bookings for a specific fitness class.
* Parameters:

class\_id: Identifier of the class to retrieve details for.

* Returns: Dictionary containing user bookings for the specified class, raises ValueError if the class is not found.

**Unit Test Cases**

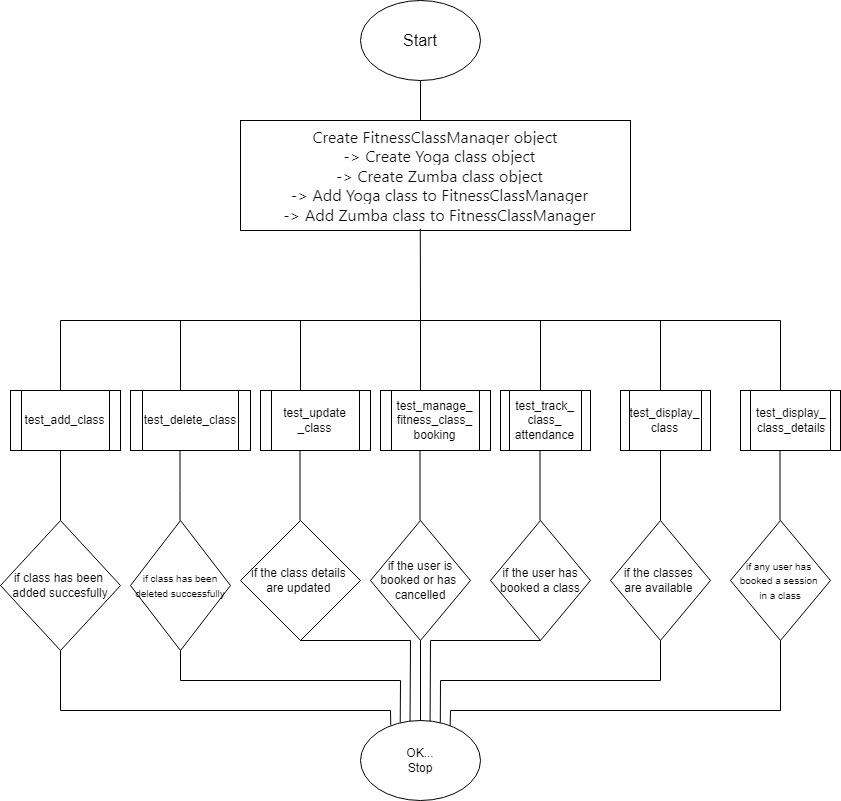
* Purpose:
* This module contains a suite of unit tests designed to verify the correct behavior of the functionalities within the fitness class booking system.
* **test\_add\_class:**
* Purpose: Tests the addition of a new fitness class to the manager.
* Checks if a class is successfully added and the number of classes in the manager increases.
* Verifies that an error is raised when attempting to delete a non-existing class.
* **test\_delete\_class:**
* Purpose: Tests the deletion of a fitness class from the manager.
* Validates whether a class is correctly deleted, reducing the number of classes in the manager.
* Ensures that an error is raised when attempting to delete a non-existing class.
* **test\_update\_class:**
* Purpose: Tests the update of class details such as name and maximum capacity.
* Validates if the class details are successfully updated.
* Verifies that an error is raised when attempting to update details of a non-existing class.
* **test\_manage\_fitness\_class\_bookings:**
* Purpose: Tests the booking and canceling of class bookings.
* Checks if bookings and cancelations are processed correctly.
* Verifies that errors are raised for invalid actions or non-existing classes.

7

* **test\_track\_class\_attendance:**
* Purpose: Tests the tracking of class attendance.
* Validates if the attendance data is accurately recorded.
* Ensures that the attendance count is updated based on bookings.
* **test\_display\_classes:**
* Purpose: Tests the display of class names.
* Validates if the names of all classes managed are correctly returned.
* **test\_display\_class\_details:**
* Purpose: Tests the display of booking details for a specific class.
* Validates if the booking details for a class are accurately retrieved.
* Ensures that an error is raised when attempting to retrieve details for a non-existing class.

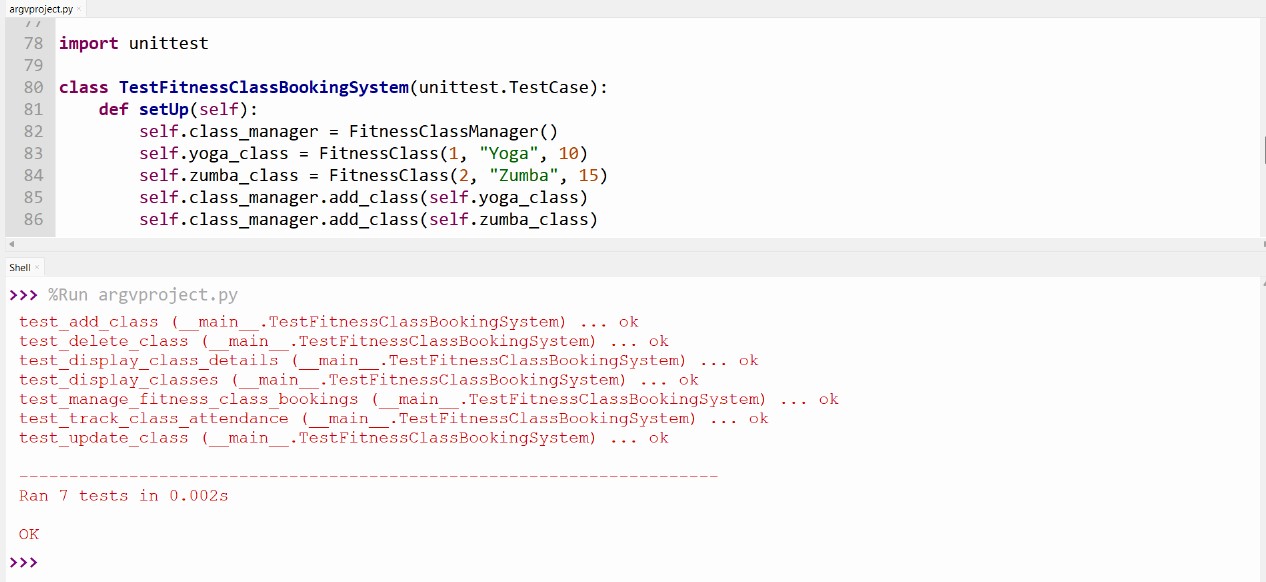
8

**FLOWCHART**

****

9

**RESULT**



10

**CONCLUSION:**

* This code creates a program for managing fitness classes for Yoga/Zumba.
* It allows Yoga/Zumba staff to add, remove, and update classes.
* Users can book and cancel their attendance for these classes, and the system keeps track of how many people attend each class.
* Additionally, the code includes tests to make sure everything works correctly.
* Overall, it's a handy tool for organizing and tracking fitness classes and attendance.

11

**REFERENCE:**

**Google Chrome**

•Chatgpt

•www.programiz.com

•https://docs.python.org/3/library/unittest.html

**Github link**

**1.** https://github.com/Sonalismudalagi/BITMS

**2.** https://github.com/ShashikalaJ/BITMS

**3.** https://github.com/3sha1234/BITMS

**4.** https://GitHub.com/kimshalu/BITMS

12