KIET GROUP OF INSTITUTIONS



# REPORT ON

TIMETABLE SHEDULER

|  |  |
| --- | --- |
| SUBMITTED BY: |  |

PRADYUMNA SARASWAT 2200290119010

PRASHUK JAIN 2200290119011

NIKUNJ GUPTA 2200290119009

YASH BAJAJ 2100290110196

# SUBMITTED TO: Mr. VINAY KUMAR

INDEX

# Introduction

* Abstract

# Objective

* Methodology

# Implementation

* Expected outcome

# Conclusion

Introduction

A time table scheduler is a tool that helps individuals and organizations efficiently manage their time by creating and organizing schedules for various activities, events, and tasks. Whether it's for school, work, or personal use, a time table scheduler can help users stay organized, prioritize tasks, and maximize productivity. By inputting important dates, deadlines, and recurring events, users can easily visualize their schedule and make adjustments as needed. With the ability to set reminders and notifications, a time table scheduler can also help users stay on track and avoid missing important appointments or deadlines. Overall, a time table scheduler is a valuable tool for effective time management and productivity.

# Abstract

In today's fast-paced world, effective time management is essential for individuals and organizations to stay organized and productive. A time table scheduler is a valuable tool that helps users create and organize schedules for various activities, events, and tasks. By inputting important dates, deadlines, and recurring events, users can easily visualize their schedule and make adjustments as needed. With the ability to set reminders and notifications, a time table scheduler also helps users stay on track and avoid missing important appointments or deadlines. Overall, a time table scheduler is an essential tool for efficient time management and maximizing productivity.

# Objective

The objective of a time table scheduler is to help individuals and organizations effectively manage their time by creating and organizing schedules for various activities, events, and tasks. It aims to provide users with a visual representation of their schedule, allowing them to make adjustments as needed and stay on track with important dates and deadlines. Additionally, the scheduler aims to help users avoid missing appointments or deadlines by providing reminders and notifications. Ultimately, the objective is to promote efficient time management and maximize productivity for users.

# Methodology

The project will involve the following steps:

1. Identify the specific needs and requirements of the individual or organization: This includes understanding the types of activities, events, and tasks that need to be scheduled, as well as any specific preferences or constraints.

2. Gather relevant information: This includes collecting details about the activities, events, and tasks that need to be scheduled, as well as any deadlines, priorities, and dependencies.

3. Choose a suitable scheduling tool: There are various software applications and tools available for creating and organizing schedules, such as digital calendars, time management apps, and project management software. Select a tool that best fits the needs and preferences of the user.

4. Input all relevant information into the scheduling tool: This includes entering details about upcoming events, appointments, deadlines, and recurring tasks. It is important to include as much detail as possible to ensure an accurate and comprehensive schedule.

5. Organize and prioritize tasks: Arrange the schedule in a logical and efficient manner, taking into account priorities, deadlines, and dependencies. This may involve categorizing tasks, assigning time slots, and allocating resources.

6. Set reminders and notifications: Utilize the features of the scheduling tool to set up reminders and notifications for important events and deadlines. This helps users stay on track and avoid missing important commitments.

7. Regularly review and update the schedule: It is important to regularly review the schedule to make adjustments as needed, such as adding new tasks, rescheduling events, or accommodating changes in priorities.

8. Monitor and track progress: Use the scheduling tool to monitor progress on tasks and activities, track time spent on different activities, and identify areas for improvement in time management.

9. Seek feedback and make improvements: Gather feedback from users about their experience with the scheduling tool and make necessary improvements to enhance its effectiveness and usability.

10. Provide support and training: Offer support and training to users on how to effectively use the scheduling tool to maximize its benefits for time management and productivity.

# Implementation

Code:

import random

import copy

class TimetableSchedulerGA:

def \_init\_(self, days, time\_slots, population\_size):

self.days = days

self.time\_slots = time\_slots

self.population\_size = population\_size

self.population = []

# Initialize a random population

for \_ in range(population\_size):

individual = {day: {time: None for time in time\_slots} for day in days}

self.randomly\_schedule\_events(individual)

self.population.append(individual)

def randomly\_schedule\_events(self, individual):

# Randomly assign events to time slots in the individual

events = ["Event1", "Event2", "Event3", "Event4", "Event5"] # Placeholder event names

for day in self.days:

for time in self.time\_slots:

if random.random() < 0.5: # 50% probability to schedule an event

event = random.choice(events)

individual[day][time] = event

def fitness(self, individual):

# Placeholder fitness function (you can replace it with a more sophisticated one)

# For simplicity, this function counts the number of unique events scheduled

scheduled\_events = set()

for day in self.days:

for time in self.time\_slots:

event = individual[day][time]

if event:

scheduled\_events.add(event)

return len(scheduled\_events)

def crossover(self, parent1, parent2):

# One-point crossover

crossover\_point = random.choice(self.days)

child = copy.deepcopy(parent1)

for day in self.days:

if day > crossover\_point:

child[day] = copy.deepcopy(parent2[day])

return child

def mutate(self, individual, mutation\_rate):

# Randomly mutate events in the individual with a certain probability

for day in self.days:

for time in self.time\_slots:

if random.random() < mutation\_rate:

individual[day][time] = None

def evolve(self, generations, mutation\_rate):

for generation in range(generations):

# Evaluate fitness of each individual in the population

fitness\_scores = [self.fitness(individual) for individual in self.population]

# Select parents for crossover based on fitness

parents = random.choices(self.population, weights=fitness\_scores, k=self.population\_size)

# Create new generation through crossover and mutation

new\_population = []

for i in range(0, self.population\_size, 2):

parent1, parent2 = parents[i], parents[i + 1]

child1 = self.crossover(parent1, parent2)

child2 = self.crossover(parent2, parent1)

self.mutate(child1, mutation\_rate)

self.mutate(child2, mutation\_rate)

new\_population.extend([child1, child2])

self.population = new\_population

# Display the best individual in the current generation

best\_individual = max(self.population, key=self.fitness)

print(f"Generation {generation + 1}, Best Fitness: {self.fitness(best\_individual)}")

self.display\_schedule(best\_individual)

def display\_schedule(self, individual):

print("\nTimetable Schedule:")

for day in self.days:

print(f"\n{day}:")

for time in self.time\_slots:

event = individual[day][time]

print(f" {time}: {event if event else 'Free'}")

def main():

# Define the days and time slots for the timetable

days = ["Monday", "Tuesday", "Wednesday", "Thursday", "Friday"]

time\_slots = ["9:00 AM", "10:00 AM", "11:00 AM", "12:00 PM", "1:00 PM", "2:00 PM", "3:00 PM", "4:00 PM"]

# Set the population size and number of generations

population\_size = 10

generations = 50

# Set the mutation rate

mutation\_rate = 0.1

# Create a timetable scheduler using a genetic algorithm

scheduler = TimetableSchedulerGA(days, time\_slots, population\_size)

# Evolve the population through generations

scheduler.evolve(generations, mutation\_rate)

# Display the final schedule

best\_individual = max(scheduler.population, key=scheduler.fitness)

print("\nFinal Timetable Schedule:")

scheduler.display\_schedule(best\_individual)

if \_name\_ == "\_main\_":

    main()

# Expected Outcome

The expected outcome for a time table scheduler is to effectively and efficiently manage and organize activities, events, and tasks, leading to improved time management, productivity, and overall performance. By using a scheduling tool, individuals and organizations can expect to have a clear and comprehensive schedule that helps them stay on track, meet deadlines, and prioritize tasks effectively. The scheduling tool should also provide reminders and notifications to prevent missed commitments, as well as the ability to monitor progress and make adjustments as needed. Ultimately, the goal of a time table scheduler is to streamline and optimize time management, leading to increased efficiency and success in achieving goals and objectives.

# Conclusion

In conclusion, a time table scheduler is a valuable tool for individuals and organizations to effectively manage their time and activities. By providing a clear and comprehensive schedule, reminders, and the ability to monitor progress, a scheduling tool can greatly improve time management, productivity, and overall performance. With the ability to prioritize tasks, meet deadlines, and make adjustments as needed, a time table scheduler is essential for streamlining and optimizing time management, ultimately leading to increased efficiency and success in achieving goals and objectives. Overall, a time table scheduler is an essential tool for anyone looking to effectively manage their time and maximize their productivity.

# Recommendations

It is recommended to conduct thorough testing and user feedback sessions to ensure the usability and effectiveness of the wireless sound control system. Additionally, future enhancements could include integration with smart devices and voice recognition capabilities for an even more intuitive user experience.

Overall, the mini project on wireless sound control holds promise for addressing the evolving needs of audio control in today's digital age.