FORM-3 METHODOLOGY

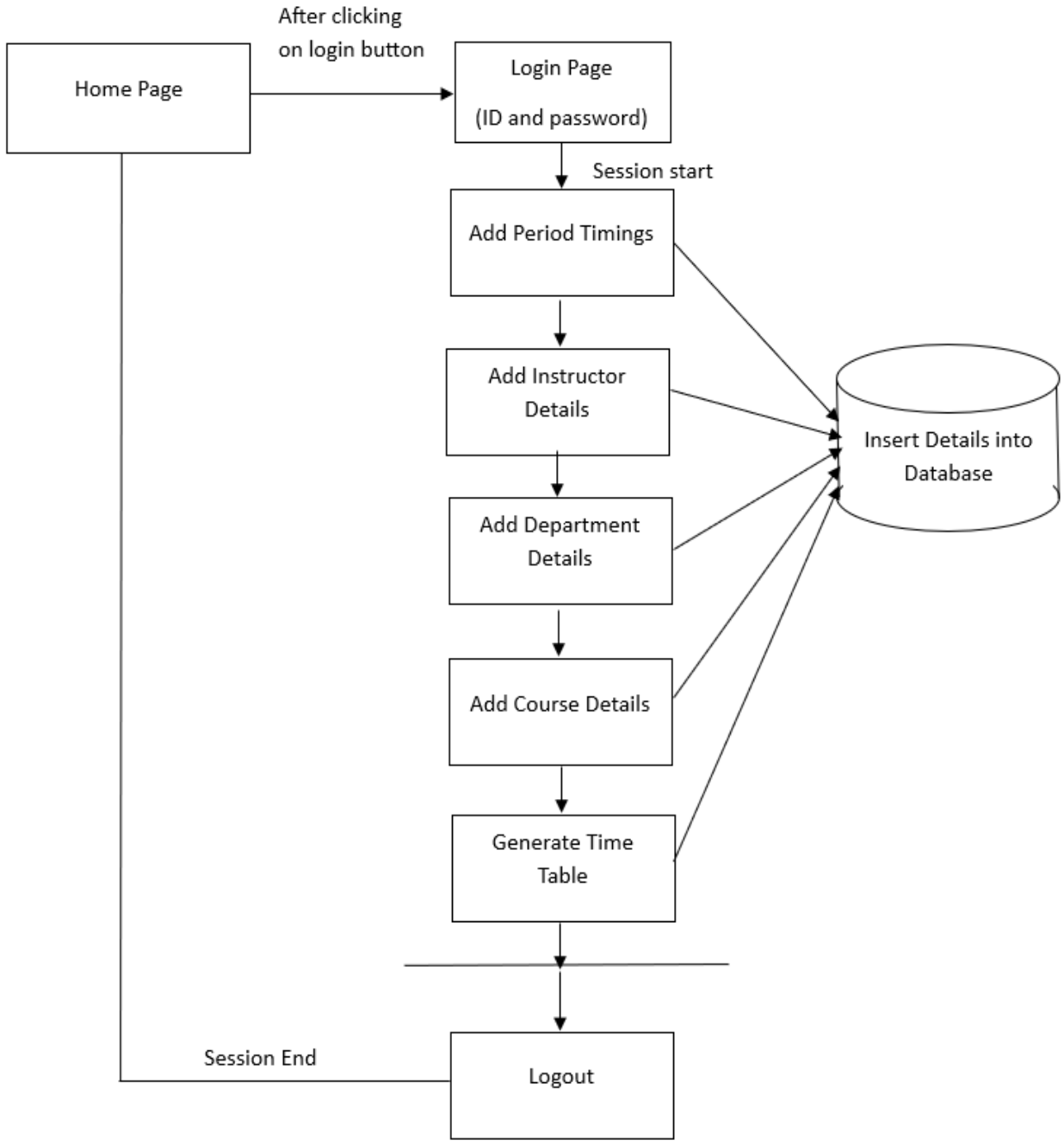
Team No: 16

Project Title: AUTOMATIC TIMETABLE GENERATION

Proposed Method:

The proposed system for Automatic timetable Generation utilizes genetic algorithms to efficiently generate optimal timetables for educational institutions. It features a user-friendly interface for inputting constraints such as teacher availability, course prerequisites, and classroom availability. The genetic algorithm creates a population of possible timetables, employing crossover and mutation operations to produce new generations. A fitness function evaluates each timetable based on constraints and objectives, selecting the fittest for the next generation. Users can visualize, evaluate, and refine schedules, and can adjust limits as needed. The system handles complex constraints, offers flexibility, and is designed for ease of use, saving time and effort for educational institutions.

Proposed Method illustration:



1. Initialization:

- Generate an initial population of chromosomes randomly, each representing a potential timetable.
- Evaluate the fitness of each chromosome based on adherence to constraints and preferences.

2. Genetic Operations:

a. Elitism:

- Preserve fittest chromosomes from the previous generation.

b. Selection:

- Randomly select chromosome pairs from the previous generation using Roulette Wheel Selection.

c. Crossover:

- Apply Single Point Crossover, ensuring it doesn't intersect student group timetables.

d. Mutation:

- Apply Swap Mutation to the more fit chromosome.

3. Fitness Evaluation of New Generation:

- Analyse fitness of the newly created chromosomes in the current generation.

4. Sorting and Selection:

- Order chromosomes based on fitness.
- Repeat until chromosomes with desired fitness value (e.g., fitness = 1) are obtained.

5. Output:

- The final selected chromosome is the optimized timetable satisfying hard and soft constraints.

6. Testing and Console Output:

- Print console information, including input data, generated slots, chromosomes, fitness values, and the final selected chromosome.

7. Conclusion and Further Work:

- Conclude the automated Time Table generation.
- Evaluate the effectiveness of Genetic Algorithm in obtaining superior solutions.
- Suggest further work: consider soft constraints like faculty preferences, classroom size, lab facilities, and multiple subject selections for faculty.

8. Technical Description:

- Technologies Used:
- Backend: Java 8, JSP, Servlets.
- Database: MySQL.
- Frontend: HTML5, CSS, JavaScript, Bootstrap, Ajax.
- Local Hosting: Apache Tomcat-7.

Parameters:

The parameter formulas can be derived from the explanation of the Genetic Algorithm (GA) process for Time Table Scheduling. Here are some of the key parameters mentioned:

1. Initialization:

- Population Size (N): Number of chromosomes in the initial population.

2. Genetic Operations:

a. Elitism:

- Elitism Ratio (ER): Percentage of fittest chromosomes preserved from the previous generation.

b. Selection:

- Selection Probability (SP): Probability of selecting a chromosome based on its fitness during the Roulette Wheel Selection.

c. Crossover:

- Crossover Rate (CR): Probability of crossover occurring between two selected chromosomes.
- Crossover Point (CP): Randomly chosen point for the Single Point Crossover, ensuring it doesn't intersect student group timetables.

d. Mutation:

- Mutation Rate (MR): Probability of mutation occurring on the more fit chromosome.
- Swap Mutation: Specific technique involving the exchange of two portions of the chromosome.

3. Sorting and Selection:

- No specific parameters, but it involves ordering chromosomes based on their fitness.

Signature of the Supervisor