Goal

- _nameGoal : string
- _descriptionGoal : string
- _pointsPerGoal : int
- _pointsEarneds : int
- _levels : List<string>

+ Goal(string name, string description, int points) : void
+ GetNameGoal() : string
- GetDescriptionGoal() : string
- SetPoints() : void
- SetBonusPoints(int points) : void
+ GetEarnedPoints() : int
- GetPointsPerGoal() : int
+ *DisplayGoal() : void ABSTRACT*
+ *RecordedEvent() : void VIRTUAL*
- *IsComplete() : void ABSTRACT*

Goal Manager

List<Goal> goals : Goal
_filename : string
_totalPoints : int

GoalManager()
RecordEvent(int position) : void
ShowGoals()
ShowNameGoals() : void
ShowScore() : void
SaveGoal(Goal goal) void
SaveGoalInFile(Goal goal) void
LoadGoalsFromFile(Goal goal) void

SimpleGoal

- _checked : bool

SimpleGoal(name, description, points, checked) : void
+ DisplayGoal() : override
+ RecordEvent() : override
- IsComplete() : overridel

CheckListGoal

- _bonusPoints : int
- _targetCount : int
- _currentCount : int
- _completed : bool

+ CheckListGoal(string name, description, int points, int bonusPoints, int targetCount)
+ RecordEvent() : override
+ DisplayScore() : void
- IsComplete() : bool

EternalGoal

+ EternalGoal()
+ RecordEvent() : override
+ DisplayScore() : void
- IsComplete() : bool

Diagram description

Goal is an abstract class that serves as a base class for representing a goal. It has properties and methods common to all goals, such as name, description, earned points, etc. It also has abstract methods like DisplayGoal and RecordEvent, which must be implemented by derived classes.

CheckListGoal, **SimpleGoal**, and **EternalGoal** are derived classes of **Goal** that represent specific types of goals. Each of these classes implements the abstract methods of the base class according to their needs. For example, **CheckListGoal** implements the IsComplete method to check if the checklist goal has been achieved and grant bonus points if necessary.

GoalManager is a class that manages and performs operations on goals. It has a list of goals and provides methods to add new goals, display existing goals, register events related to goals, calculate the total score, etc. It uses instances of goal classes to perform these operations.

The interaction between these classes is primarily through the instance of GoalManager. When creating a new goal, an instance of the corresponding class (e.g., SimpleGoal, EternalGoal, or CheckListGoal) is created and saved in the goal list of GoalManager. When registering events related to goals, the appropriate method in the corresponding goal is invoked through the instance of GoalManager. Additionally, GoalManager uses methods and properties of goals to perform operations such as displaying existing goals and calculating the total score.

In summary, the classes interact through the instance of GoalManager, which acts as a central point for managing and operating on goals. The derived classes of Goal represent different types of goals and provide specific implementations of the abstract methods of the base class. GoalManager uses instances of these derived classes to perform operations related to goals.