```csharp
        /// <summary>
        /// get BattleAction choice from the Dalek and the Traveler
        /// calculate the BattleResult based on the BattleActions and
        /// return it
        /// </summary>
        /// <returns>Battle Result</returns>
        private BattleResult Battle()
        {
            BattleResult battleResult = BattleResult.None;
            Dalek daleckToBattle;
            BattleAction travelerBattleActionChoice;
            BattleAction dalekBattleActionChoice;
            Random random = new Random();

            daleckToBattle = _gameConsoleView.DisplayGetDalekToBattle();

            if (daleckToBattle != null)
            {
                travelerBattleActionChoice = _gameConsoleView.DisplayGetBattleActionChoice();
                dalekBattleActionChoice = daleckToBattle.GetBattleAction();

                if (travelerBattleActionChoice == BattleAction.Retreat
                        & dalekBattleActionChoice == BattleAction.Retreat)
                {
                    battleResult = BattleResult.BothRetreat;
                }
                else if (travelerBattleActionChoice == BattleAction.Attack
                            & dalekBattleActionChoice == BattleAction.Retreat)
                {
                    battleResult = BattleResult.NPCRetreats;
                }
                else if (travelerBattleActionChoice == BattleAction.Retreat
                            & dalekBattleActionChoice == BattleAction.Attack)
                {
                    battleResult = BattleResult.TravelerRetreats;
                }
                else if (travelerBattleActionChoice == BattleAction.Attack
                            & dalekBattleActionChoice == BattleAction.Attack)
                {
                    int travelerBattleNumber = random.Next(1, 100) * _gameTraveler.BattleIndex;
                    int dalekBattleNumber = random.Next(1, 100) * daleckToBattle.BattleIndex;

                    if (travelerBattleNumber > dalekBattleNumber)
                    {
                        battleResult = BattleResult.TravelerWins;
                    }
                    else
                    {
                        battleResult = BattleResult.NPCWins;
                    }
                }
            }

            ProcessBattleResult(battleResult, daleckToBattle);

            return battleResult;
        }

        /// <summary>
        /// perform the required tasks based on the battle results
        /// </summary>
        /// <param name="battleResult">result of the battle</param>
        /// <param name="daleckToBattle">dalek battled</param>
        private void ProcessBattleResult(BattleResult battleResult, Dalek daleckToBattle)
        {
            switch (battleResult)
            {
            case BattleResult.TravelerWins:
                    daleckToBattle.SpaceTimeLocationID = 0; // Daleck removed from game
                    break;
            case BattleResult.NPCWins:
                    _gameTraveler.Lives--;
                    break;
            case BattleResult.TravelerRetreats:
            case BattleResult.NPCRetreats:
            case BattleResult.Draw:
            case BattleResult.BothRetreat:
                    break;
            default:
                    break;
            }
        }
```