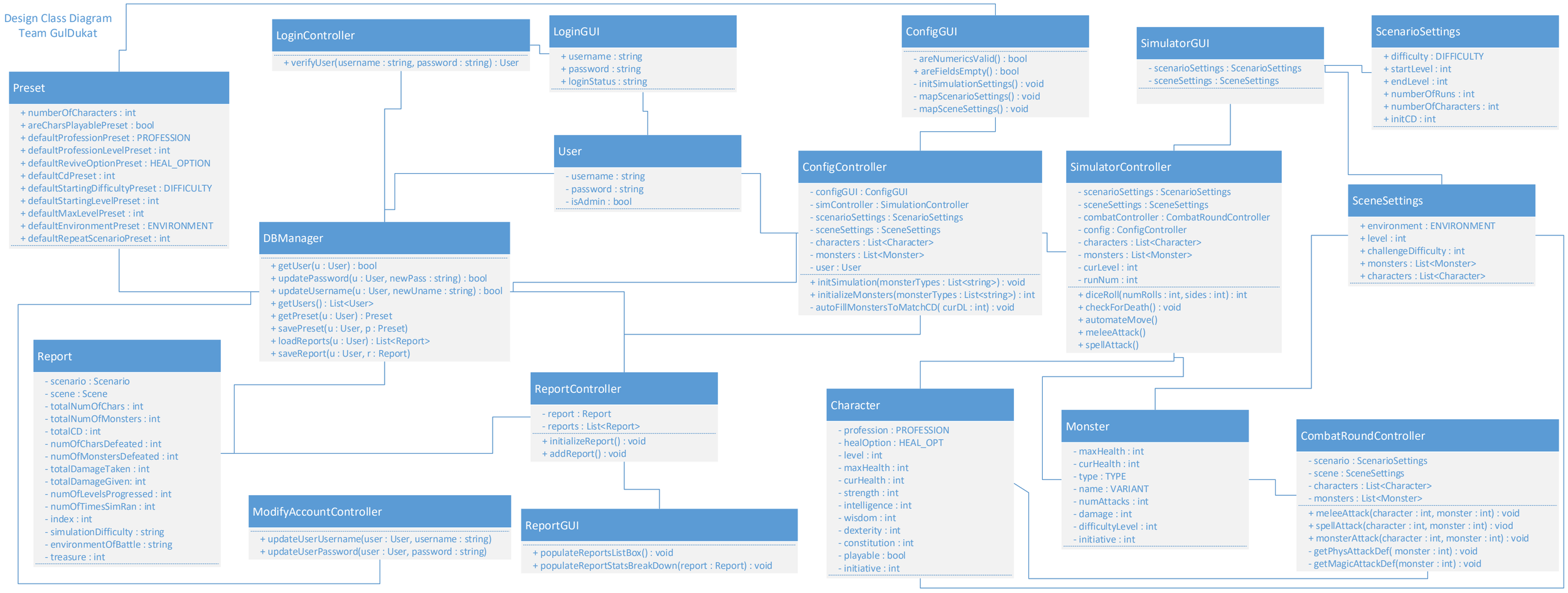
Design Class Diagram
Team GulDukat
LoginController
+ verifyUser(username : string, password : string) : User
LoginGUI
+ username : string
+ password : string
+ loginStatus : string
ConfigGUI
- areNumericsValid() : bool
+ areFieldsEmpty() : bool
- initSimulationSettings() : void
- mapScenarioSettings() : void
- mapSceneSettings() : void
SimulatorGUI
- scenarioSettings : ScenarioSettings
- sceneSettings : SceneSettings
ScenarioSettings
+ difficulty : DIFFICULTY
+ startLevel : int
+ endLevel : int
+ numberOfRuns : int
+ numberOfCharacters : int
+ initCD : int
Preset
+ numberOfCharacters : int
+ areCharsPlayablePreset : bool
+ defaultProfessionPreset : PROFESSION
+ defaultProfessionLevelPreset : int
+ defaultReviveOptionPreset : HEAL_OPTION
+ defaultCdPreset : int
+ defaultStartingDifficultyPreset : DIFFICULTY
+ defaultStartingLevelPreset : int
+ defaultMaxLevelPreset : int
+ defaultEnvironmentPreset : ENVIRONMENT
+ defaultRepeatScenarioPreset : int
User
- username : string
- password : string
- isAdmin : bool
ConfigController
- configGUI : ConfigGUI
- simController : SimulationController
- scenarioSettings : ScenarioSettings
- sceneSettings : SceneSettings
- characters : List<Character>
- monsters : List<Monster>
- user : User
+ initSimulation(monsterTypes : List<string>) : void
+ initializeMonsters(monsterTypes : List<string>) : int
- autoFillMonstersToMatchCD(curDL : int) : void
SimulatorController
- scenarioSettings : ScenarioSettings
- sceneSettings : SceneSettings
- combatController : CombatRoundController
- config : ConfigController
- characters : List<Character>
- monsters : List<Monster>
- curLevel : int
- runNum : int
+ diceRoll(numRolls : int, sides : int) : int
+ checkForDeath() : void
+ automateMove()
+ meleeAttack()
+ spellAttack()
SceneSettings
+ environment : ENVIRONMENT
+ level : int
+ challengeDifficulty : int
+ monsters : List<Monster>
+ characters : List<Character>
DBManager
+ getUser(u : User) : bool
+ updatePassword(u : User, newPass : string) : bool
+ updateUsername(u : User, newUname : string) : bool
+ getUsers() : List<User>
+ getPreset(u : User) : Preset
+ savePreset(u : User, p : Preset)
+ loadReports(u : User) : List<Report>
+ saveReport(u : User, r : Report)
Report
- scenario : Scenario
- scene : Scene
- totalNumOfChars : int
- totalNumOfMonsters : int
- totalCD : int
- numOfCharsDefeated : int
- numOfMonstersDefeated : int
- totalDamageTaken : int
- totalDamageGiven: int
- numOfLevelsProgressed : int
- numOfTimesSimRan : int
- index : int
- simulationDifficulty : string
- environmentOfBattle : string
- treasure : int
ReportController
- report : Report
- reports : List<Report>
+ initializeReport() : void
+ addReport() : void
Character
- profession : PROFESSION
- healOption : HEAL_OPT
- level : int
- maxHealth : int
- curHealth : int
- strength : int
- intelligence : int
- wisdom : int
- dexterity : int
- constitution : int
- playable : bool
- initiative : int
Monster
- maxHealth : int
- curHealth : int
- type : TYPE
- name : VARIANT
- numAttacks : int
- damage : int
- difficultyLevel : int
- initiative : int
CombatRoundController
- scenario : ScenarioSettings
- scene : SceneSettings
- characters : List<Character>
- monsters : List<Monster>
+ meleeAttack(character : int, monster : int) : void
+ spellAttack(character : int, monster : int) : viod
+ monsterAttack(character : int, monster : int) : void
- getPhysAttackDef(monster : int) : void
- getMagicAttackDef(monster : int) : void
ModifyAccountController
+ updateUserUsername(user : User, username : string)
+ updateUserPassword(user : User, password : string)
ReportGUI
+ populateReportsListBox() : void
+ populateReportStatsBreakDown(report : Report) : void