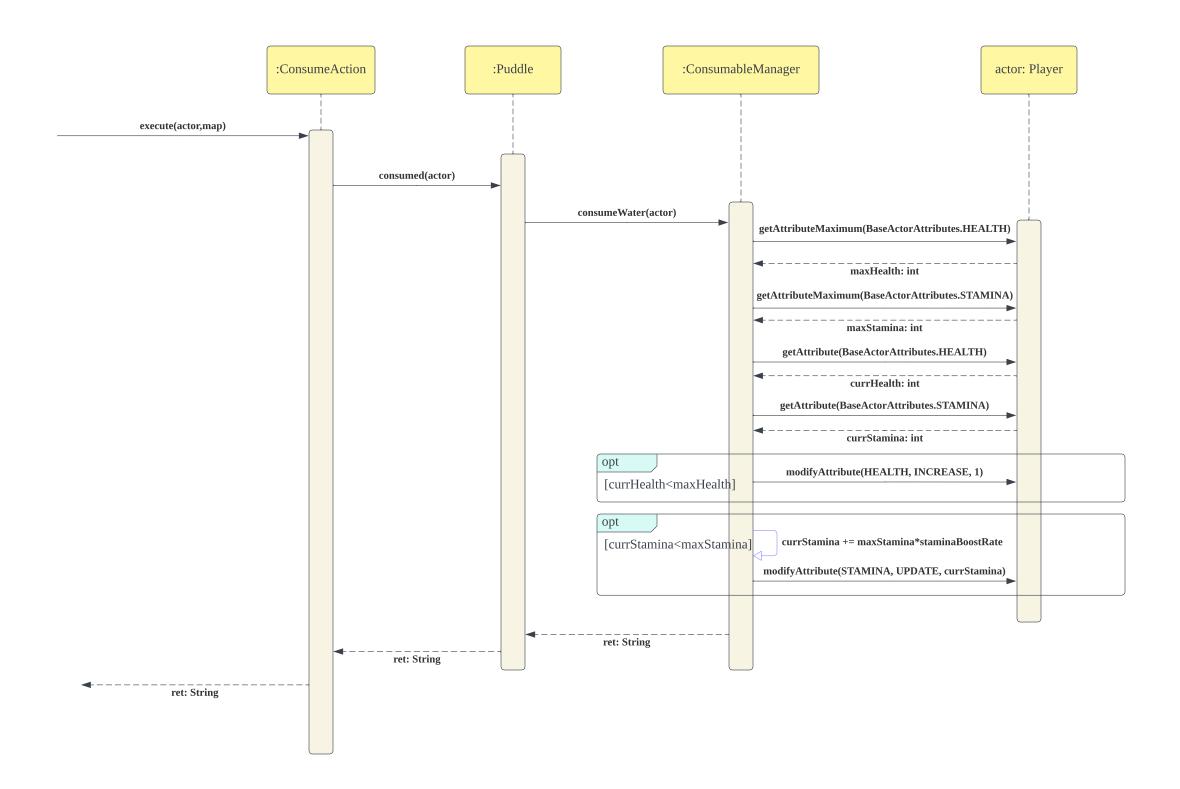
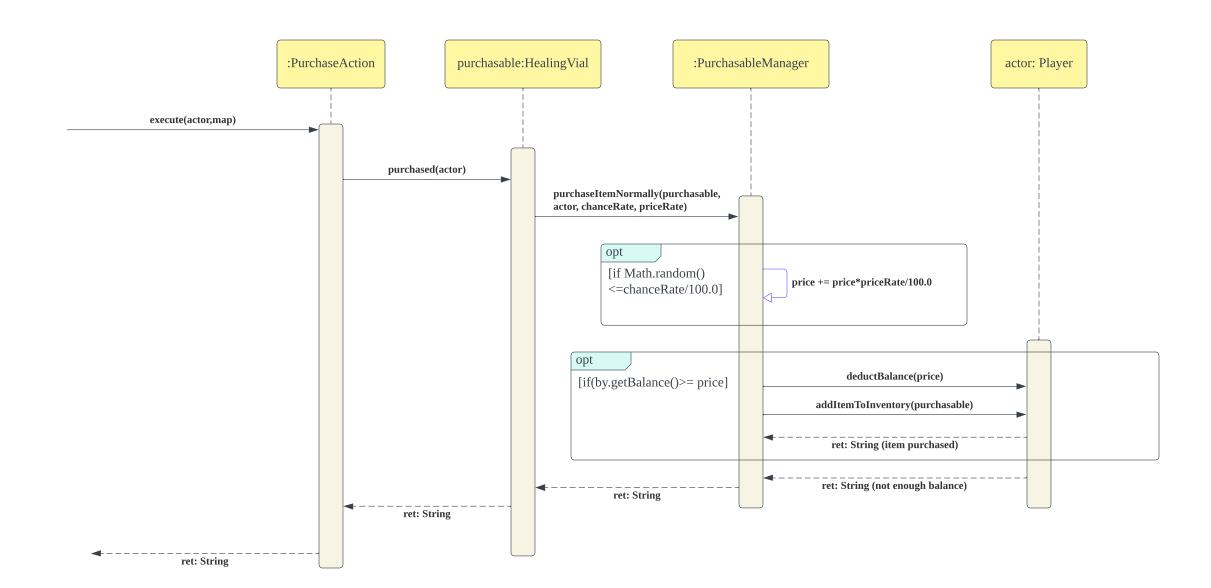
enemy:RedWolf :FollowBehaviour loop playTurn(actions, lastAction, map, display) [Behaviour behaviour: behaviours.values()] getAction(enemy,map) opt [!theAncientWoods.contains(target) || !theAncientWoods.contains(actor) theAncientWoods: GameMap actions: ArrayList<Action> <<creates>> locationOf(actor) actorLocation: Location locationOf(target) targetLocation: Location exit:Exit :MoveActorAction getDistance(actorLocation, targetLocation) loop [Exit exit:actorL<mark>ocati</mark>on.getExits()] getDestination() destination: Location [destination.ca<mark>nAc</mark>torEnter(actor)] getDistance(destination, targetLocation) [newDistance < currDistance)] <<creates>> add(moveActorAction) [!actions.isEmpty()] get(random.nextInt(actions.size())) moveActorAction:Action action: MoveActorAction [Else] [action!=null] action: MoveActorAction action: DoNothingAction

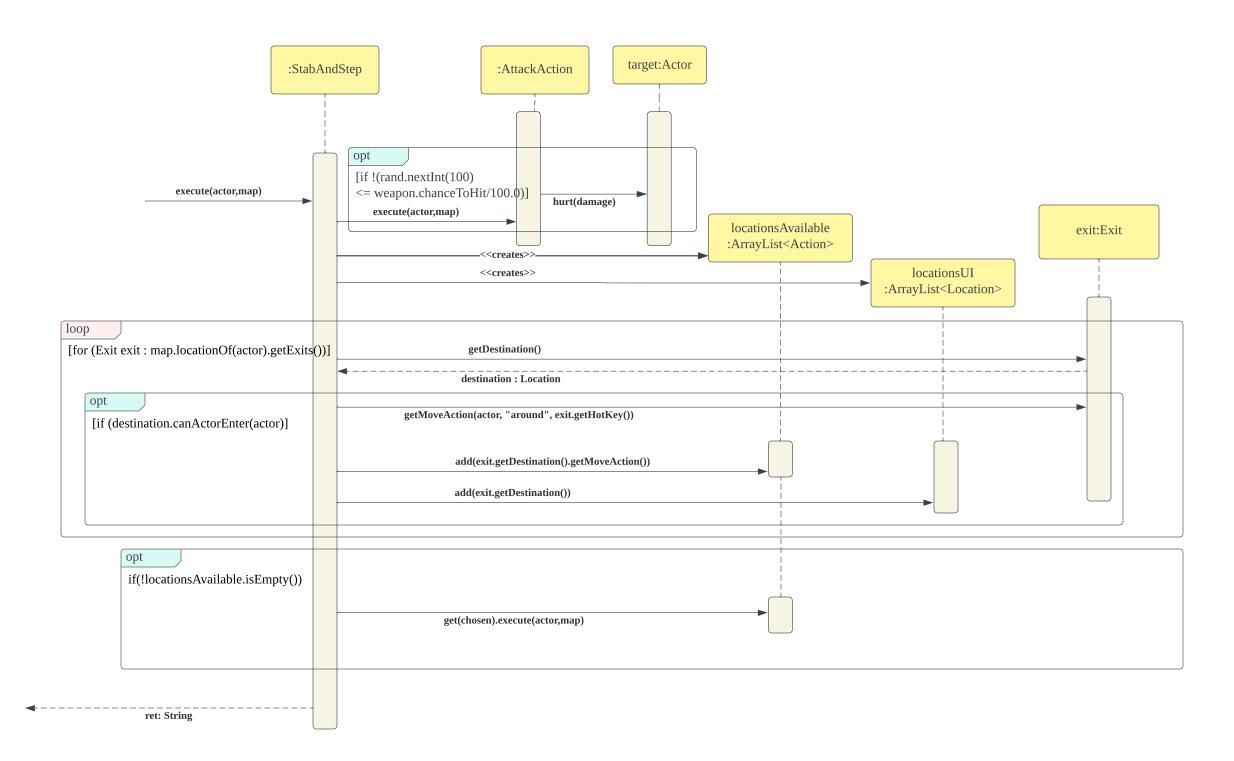
REQ1 Scenario: RedWolf follows player after being provoked

REQ2 Scenario: Player chooses to drink from puddle they are stepping on.



REQ3 Scenario: Player purchases Healing Vial from the isolated traveller





REQ5 Scenario: At its turn, the Abxervyer controls the weather

