arrows

displayAttacks(areaId)
displayArrow(mapX, mapY, unitIndex)
displayAttackOptions(id)
clearArrows()

moving

var destinationId
var originId
var moveDestinationIds
var fromAreas

---

startMove(controller)
findUnitsAreas(control)
hightlightUnitAreas(allOrigins)
addMoveOriginListeners(allOrigins)
moveOriginHandler(event)
removeOriginListeners(allOrigins)
addMoveDestinationListeners(allOrigins)
rotateUnits(targetAreaId)
rotateUnitsHandler()
moveDestinationHandler(event)
removeDestinationListeners(allOrigins
clearUnits()

> These highlighted methods are not new but have had changes made to them.

highLights

highlightArea(coords, highlightColor = PALE_ORANGE)
highlightFriendly(coords, highlightColor = LIGHT_GREY)
highlightEneamy(coords, highlightColor = LIGHT_PURPLE)
clearHighlights()

unitSetUp

var turn = 1
defaultSetUp()
displayUnits()
displayAllied(mapX, mapY, unitIndex)
displayGerman(mapX, mapY, unitIndex)

attacking

var attackingAreaId
var defendingAreaId
var allFriendlyFrontLineAreas = []
var enemyNeighbours = []

---

findFrontline(friendly)
startAttack(friendly)
hightlightEnemyUnitAreas(allOrigins)
addAttackOriginListeners(allOrigins)
attackOriginHandler(event)
removeAttackOriginListeners(allOrigins)
addAttackedAreaListeners(allOrigins)
attackDestinationHandler(event)
removeAttackDestinationListeners(allOrigins)

> Call game.addAttack()
> in attackDestinationFinder()
>
> Is there any need to change
> findFrontline() like
> findUnitsArea() in moving?

start

addSplashListerner()
removeSplashListerner()
continueCampaignHandler()
startCampaignHandler()

intergrated2

loadHandler()

Area

id:string
terrain:string
canSupplyAllied:boolean
canSupplyGerman:boolean
control:string
centerX:int
centerY:int
vp:int
fullName:string
_hasAttacked:boolean false

---

addUnit(unit)
get areaName()
addNeighbour(aNeighbour)
addAttackArrow(anAttackArrow)
findArrowOptions(id)
clearAttack()
makeAttack()

> Call game.addAttack()

Clock

_currentTurn:int
_totalTurn:int
_allDates:[]
_allRules[]
_allDescriptions:[]

---

get turn()
get date()
get specialRules()
get description()
nextTurn()

Attack

_turn
_attackingAreaId:string
_defendingAreaId:string

> Removed set methods.
> Added turn as a
> parameter to the class.

Game

_allUnits:[]
_allAreas:[]
_allMoves: []
_allAttacks:[]
_allReinforcementUnits = []
_clock

---

addUnit(id, name, army, training, branch, startTurn, startPosition)
addReinforcementUnit(id, name, army, training, branch, startTurn, startPosition)
reportAlliedUnits()
reportGermanUnits()
findAdjacentEneamyAreas(targetAreaId)
findAdjacentFriendlyAreas(targetAreaId)
addArea(id, terrain, canSupplyAllied, canSupplyGerman, control, centerX, centerY, vp = 0, fullName = "")
reportAreas()
getUnitsInArea(targetAreaId)
getArea(targetAreaId)
addNeighbours(targetAreaId, allNeighbourIds)
findAttackArrows(targetAreaId)
displayAttackArrows(attackAreaId, allNeighbourIds)
addAttackArrows(attackAreaId, allNeighbourIds)
addMove(originId, destinationId, unitId)
addAttack(originId, destinationId, areaId)

> Create addAttack()

Unit

id:string
name:string
army:string
training:string
branch:string
startTurn:int
startPosition:string
_currentPosition:string
_hasMoved :boolean false

---

get name()
get army()
clearMove()
makeMove()

Move

_turn
_originId
_destinationId
_unitId

> Removed set methods.
> Added turn as a
> parameter to the class.