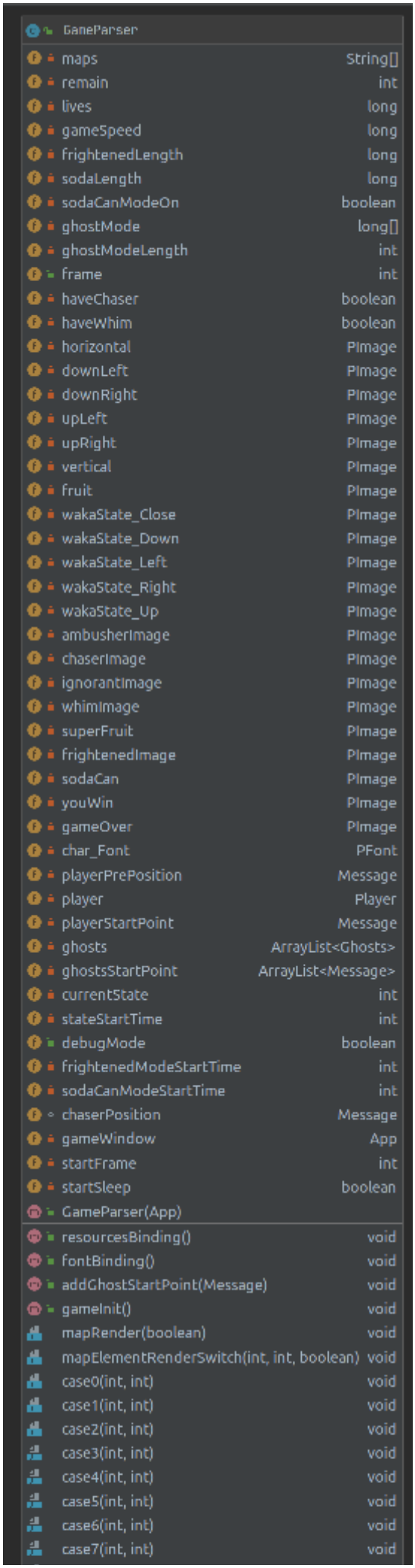
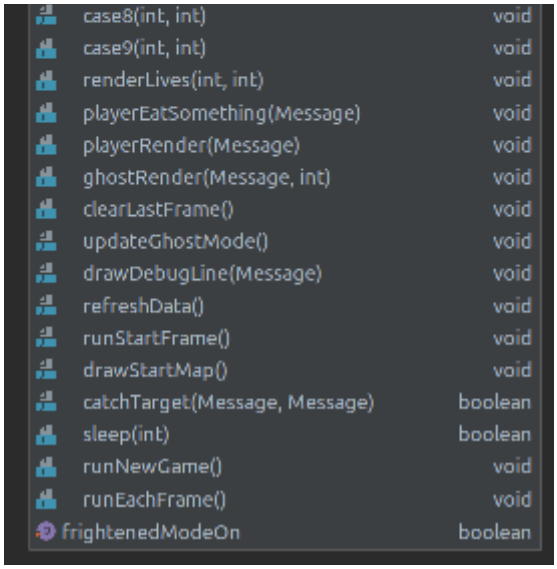
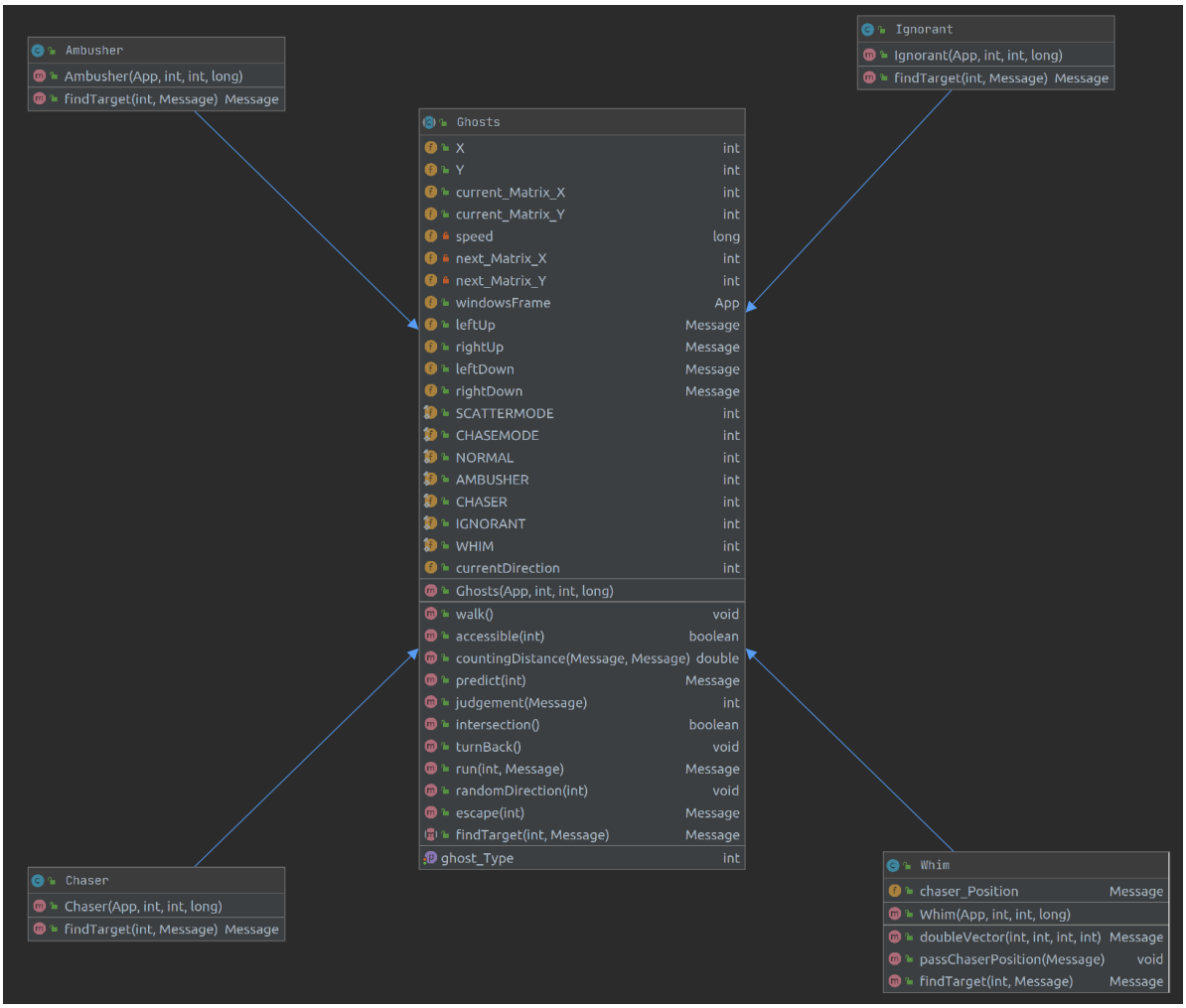
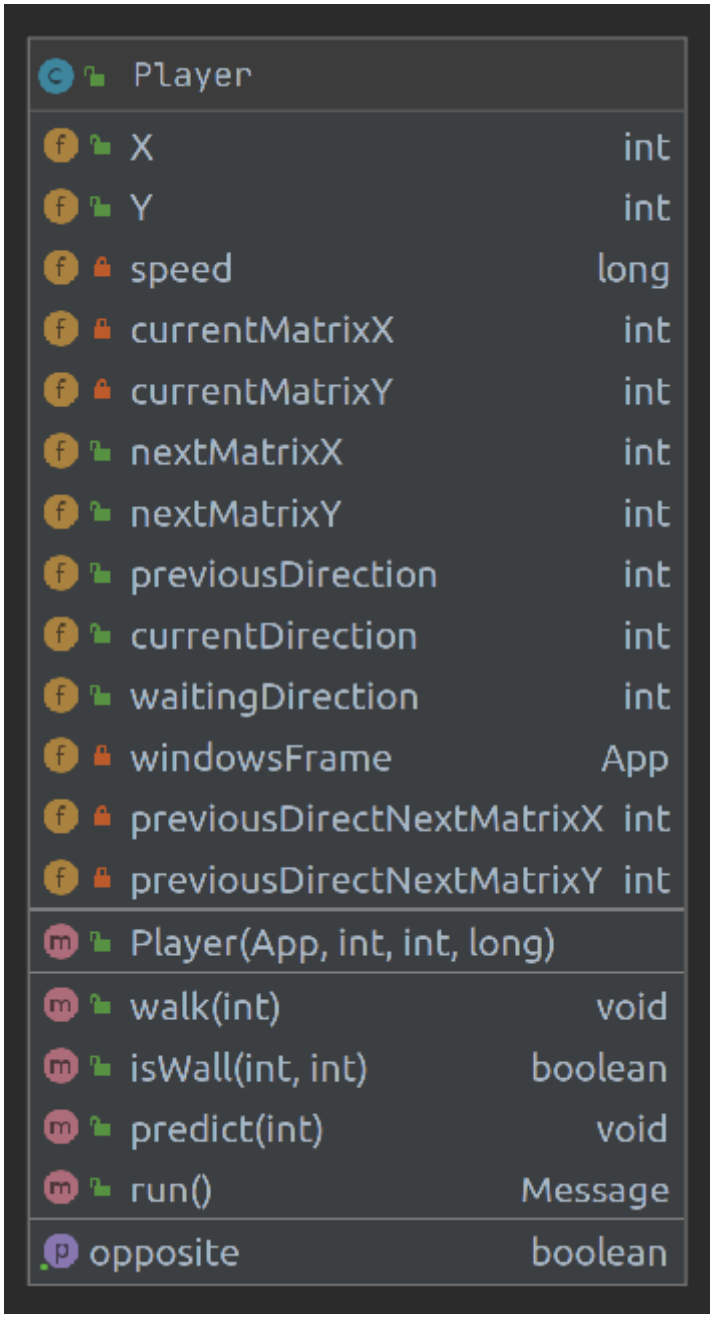
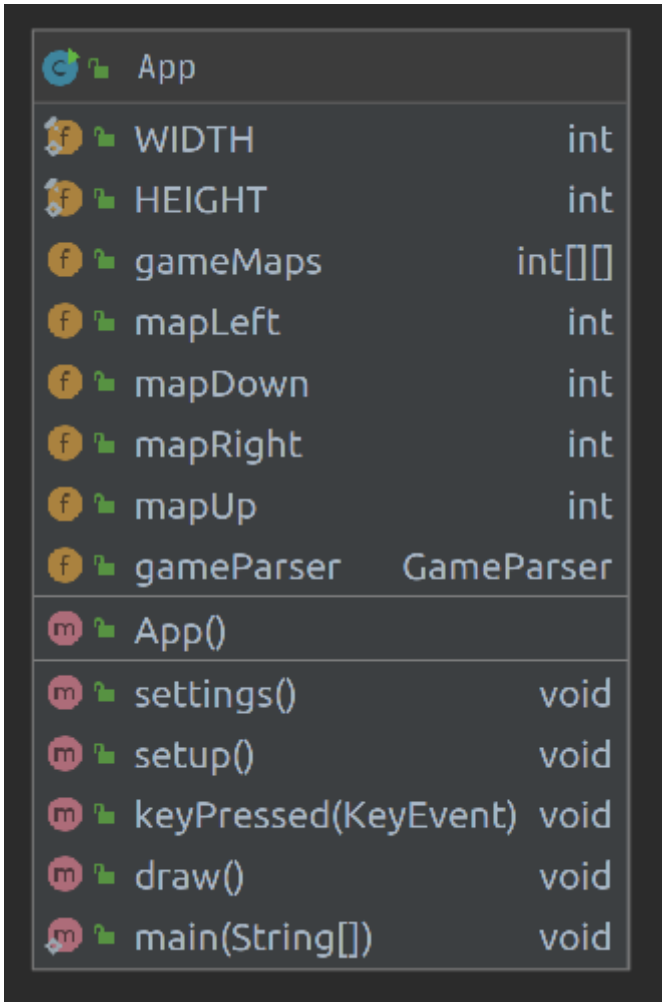
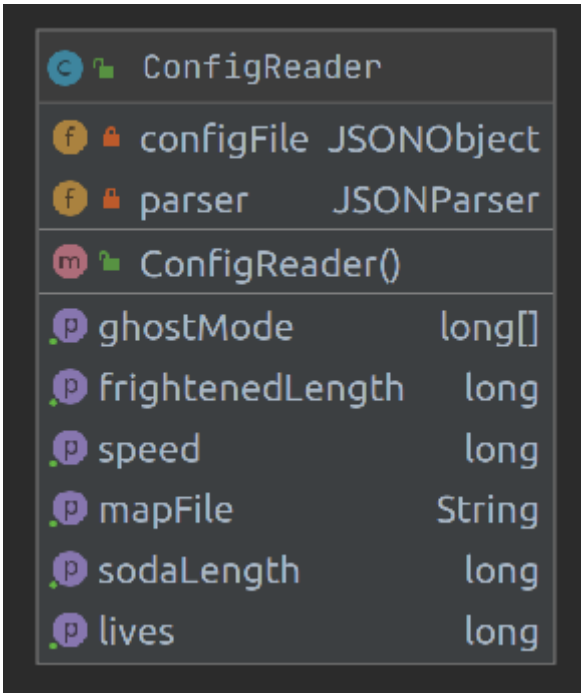

GameParser
maps String[]
remain int
lives long
gameSpeed long
frightenedLength long
sodaLength long
sodaCanModeOn boolean
ghostMode long[]
ghostModeLength int
frame int
haveChaser boolean
haveWhim boolean
horizontal PImage
downLeft PImage
downRight PImage
upLeft PImage
upRight PImage
vertical PImage
fruit PImage
wakaState_Close PImage
wakaState_Down PImage
wakaState_Left PImage
wakaState_Right PImage
wakaState_Up PImage
ambusherImage PImage
chaserImage PImage
ignorantImage PImage
whimImage PImage
superFruit PImage
frightenedImage PImage
sodaCan PImage
youWin PImage
gameOver PImage
char_Font PFont
playerPrePosition Message
player Player
playerStartPoint Message
ghosts ArrayList<Ghosts>
ghostsStartPoint ArrayList<Message>
currentState int
stateStartTime int
debugMode boolean
frightenedModeStartTime int
sodaCanModeStartTime int
chaserPosition Message
gameWindow App
startFrame int
startSleep boolean
GameParser(App)
resourcesBinding() void
fontBinding() void
addGhostStartPoint(Message) void
gameInit() void
mapRender(boolean) void
mapElementRenderSwitch(int, int, boolean) void
case0(int, int) void
case1(int, int) void
case2(int, int) void
case3(int, int) void
case4(int, int) void
case5(int, int) void
case6(int, int) void
case7(int, int) void

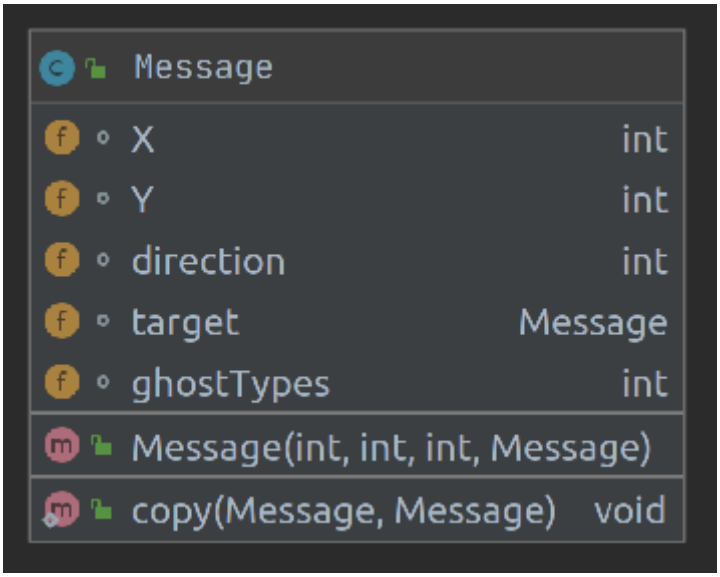

case8(int, int) void
case9(int, int) void
renderLives(int, int) void
playerEatSomething(Message) void
playerRender(Message) void
ghostRender(Message, int) void
clearLastFrame() void
updateGhostMode() void
drawDebugLine(Message) void
refreshData() void
runStartFrame() void
drawStartMap() void
catchTarget(Message, Message) boolean
sleep(int) boolean
runNewGame() void
runEachFrame() void
frightenedModeOn boolean


Ambusher
Ambusher(App, int, int, long)
findTarget(int, Message) Message
Ignorant
Ignorant(App, int, int, long)
findTarget(int, Message) Message
Ghosts
X int
Y int
current_Matrix_X int
current_Matrix_Y int
speed long
next_Matrix_X int
next_Matrix_Y int
windowsFrame App
leftUp Message
rightUp Message
leftDown Message
rightDown Message
SCATTERMODE int
CHASEMODE int
NORMAL int
AMBUSHER int
CHASER int
IGNORANT int
WHIM int
currentDirection int
Ghosts(App, int, int, long)
walk() void
accessible(int) boolean
countingDistance(Message, Message) double
predict(int) Message
judgement(Message) int
intersection() boolean
turnBack() void
run(int, Message) Message
randomDirection(int) void
escape(int) Message
findTarget(int, Message) Message
ghost_Type int
Chaser
Chaser(App, int, int, long)
findTarget(int, Message) Message
Whim
chaser_Position Message
Whim(App, int, int, long)
doubleVector(int, int, int, int) Message
passChaserPosition(Message) void
findTarget(int, Message) Message


Player
X int
Y int
speed long
currentMatrixX int
currentMatrixY int
nextMatrixX int
nextMatrixY int
previousDirection int
currentDirection int
waitingDirection int
windowsFrame App
previousDirectNextMatrixX int
previousDirectNextMatrixY int
Player(App, int, int, long)
walk(int) void
isWall(int, int) boolean
predict(int) void
run() Message
opposite boolean


App
WIDTH int
HEIGHT int
gameMaps int[][]
mapLeft int
mapDown int
mapRight int
mapUp int
gameParser GameParser
App()
settings() void
setup() void
keyPressed(KeyEvent) void
draw() void
main(String[]) void


ConfigReader
configFile JSONObject
parser JSONParser
ConfigReader()
ghostMode long[]
frightenedLength long
speed long
mapFile String
sodaLength long
lives long


Message
X int
Y int
direction int
target Message
ghostTypes int
Message(int, int, int, Message)
copy(Message, Message) void