ANEXO B: Diagrama de clases extensión FAIA frsf.cidisi.faia.agent.Perception frsf.cidisi.faia.agent.search.SearchBasedAgent frsf.cidisi.faia.agent.search.GoalTest PatrulleroPerception Patrullero **ObjetivoPatrullero** -obstaculos detectables: Set<Obstaculo> -estrategia: Estrategias De Busqueda -tipolncidente: Tipolncidente +isGoalState(AgentState agentState): boolean -selectedAction: Action -avisarCambios +selectAction(): Action <<Interfaz>> frsf.cidisi.faia.solver.search.lEstimatedCostFunction +see(p: Perception) ChangeListenerPatrullero +getAgentState(): EstadoPatrullero +getGoalString(): String +getSelectedActionStr(): String +cambio(): Boolean 1...1 +fuisteExitoso() +finSimulacionExitosa() Heuristic +noFuisteExitoso() +finSimulacionNoExitosa() +getEstimatedCost(node: NTree): double frsf.cidisi.faia.state.EnvironmentState frsf.cidisi.faia.solver.search.IStepCostFunction frsf.cidisi.faia.agent.search.SearchBasedAgentState EstadoAmbiente CostFunction **EstadoPatrullero** -mapa: Mapa +calculateCost(node: NTree): double -ultimaCalleRecorridaAgente: Arista -mapa: Mapa -posicionAgente: Interseccion -ultimaCalleRecorrida: Arista -orientacionAgente: ListIterator<Arista> frsf.cidisi.faia.environment.Environment -posicion: Interseccion -obstaculos: Set<Obstaculo> -incidente: Interseccion -hora: Long -orientacion: ListIterator<Arista> -agenteEnCorteTotal: Boolean -obstaculos: Map<Visibilidad, Set<Obstaculo>> +getObstaculosVisiblesAgente(): Collection<Obstaculo> +clone(): EstadoPatrullero **AmbienteCiudad** +addHora(cost: Long) +updateState(p: Perception) +initOrientacion() +initOrientacion() +getEnvironmentState(): EstadoAmbiente +getPercept(): PatrulleroPerception frsf.cidisi.faia.agent.search.SearchAction +agentFailed(actionReturned: Action): boolean Avanzar +execute(s: SearchBasedAgentState): SearchBasedAgentState CambiarOrientacion +execute(ast: AgentState, est: EnvironmentState): EnvironmentState -getCostAvanzar(sbs: SearchBasedAgentState): Integer -getCost(sbs: EnvironmentState): Integer +execute(s: SearchBasedAgentState): SearchBasedAgentState

-getCosto(interseccion: Interseccion, arista: Arista, hora: Long): Integer

+getCost(sbs: SearchBasedAgentState): Double

+execute(ast: AgentState, est: EnvironmentState): EnvironmentState

+getCost(sbs: SearchBasedAgentState): Double