

SysML: SINCRO



Unidade curricular: Arquitetura de Sistemas Informáticos

Professor: Luís Osório

Alunos:

André Grilo nº 46440

Valter Francisco nº 39383

Índice

1. Introdução	3
2. Apresentação do trabalho	4
3. Conclusão	14
4. Bibliografia	15

1. Introdução

Para este segundo trabalho proposto pelo docente, foi modelada a rede portuguesa de controlo de velocidade denominada de SINCRO, cujo responsável é a ANSR (Autoridade Nacional de Segurança Rodoviária). Esta rede é responsável pelo controlo, em zonas previamente assinaladas, do excesso de velocidade dos veículos cuja passagem é observada, por forma a prevenir acidentes, e a promover a atenção dos condutores aquando a sua condução.

Esta rede, SINCRO, é composta por um sistema central SIGET (Sistema de Gestão de Eventos de Trânsito) que tem por função recolher as ocorrências, processá-las, e se de acordo com o previsto, enviá-las para as entidades competentes, de forma a dar seguimento ao processo de contraordenação. Por sua vez, existe o sistema SCoT (Sistema de Contraordenações), que fornece informações ao anterior como a matrícula, a marca, o modelo, a cor e o proprietário do veículo. Estas informações são extraídas de um sistema informático do IMT (Instituto da Mobilidade e dos Transportes). [2]

Assim, o sistema SCoT apresenta-se como um subsistema do sistema SIGET, que obtém informações de outro sistema (IMT). [2]

Na componente física, a rede SINCRO é assim constituída pelos postos de controlo, que possuem uma cabine. Cabine essa que poderá conter ou não um cinemómetro, responsável por registar, com recurso à fotografia, os eventos de trânsito, e que se encontra introduzido numa gaveta standardizada para o uso desse componente caso seja de diferentes fornecedores; um acelerómetro, sensores de temperatura, de intrusão e de presença do referido cinemómetro. [2]

Baseámo-nos no guião MagicGrid para formular esta rede, que irá apresentar todas as dependências, operações, interligações e responsabilidades inerentes a este sistema que se apresenta como complexo, como se pode ver, fazendo uso do cruzamento de dados entre as imagens captadas, a velocidade a que se dirigia o veículo, a velocidade permitida na zona de localização deste ponto de controlo e as informações referentes ao veículo.

2. Apresentação do trabalho

Baseado na interpretação do documento fornecido pelo docente o trabalho foi construído com intuito de monitorizar eventos de excesso de velocidade.

Tendo isso em conta iniciámos a construção do guião MagicGrid.

	PILLAR			
		REQUIREMENTS	BEHAVIOR	STRUCTURE
DOMAIN	PROBLEM (BLACK BOX)	SINCRO	B2 Use Cases	B3 System Context
	PROBLEM (WHITE BOX)		W2 Functional Analysis	W3 Logical Subsystem Communication
	SOLUTION	S1 System Requirements	S2 System Behavior	S3 System Structure
		SS1 Subsystem Requirements	SS2 Subsystem Behavior	SS3 Subsystem Structure
		C1 Component Requirements	C2 Component Behavior	C3 Component Structure

Imagem 1 - MagicGrid

Após fazer uma pesquisa sobre o MagicGrid percebemos que deveríamos começar por um diagrama de requerimentos. [1]

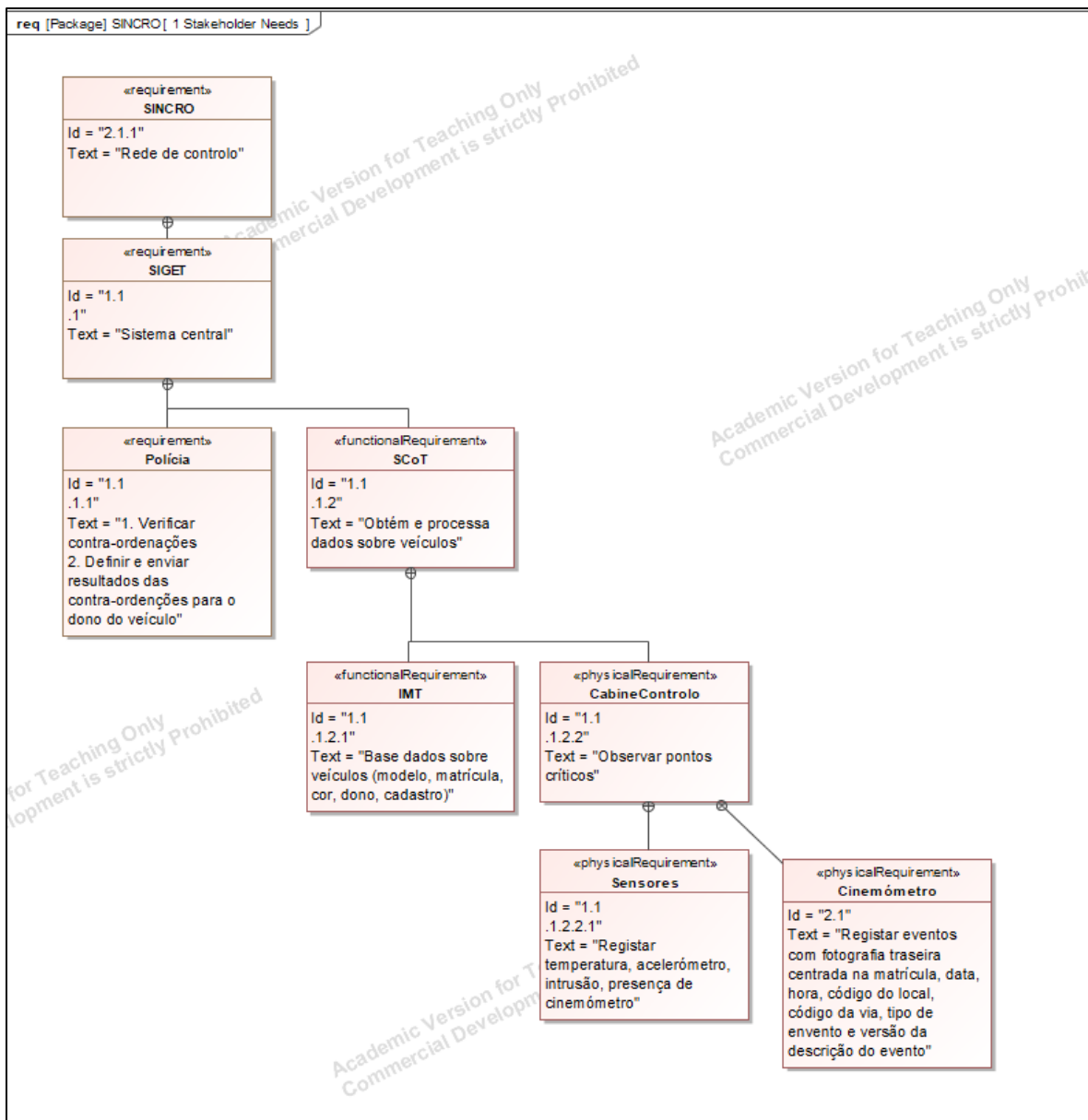


Imagem 2 - Stakeholder Needs

Começando por uma abordagem mais geral de como os intervenientes estão ligados construímos o diagrama anterior, mostrando de certa forma a hierarquia e uma linha de dependência entre estes, respondendo à primeira caixa do MagicGrid a que demos o nome de SINCRO, visto que são os stakeholder needs do sistema.

Seguidamente fizemos uma análise sobre quem iria usar o sistema usando um Use Case Diagram e criando um Activity Diagram com as interações mais básicas, que remete ao comportamento (Behavior) do sistema.

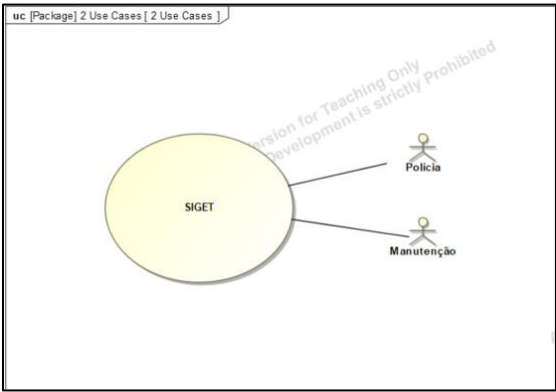


Imagem 3 - Use Case diagram

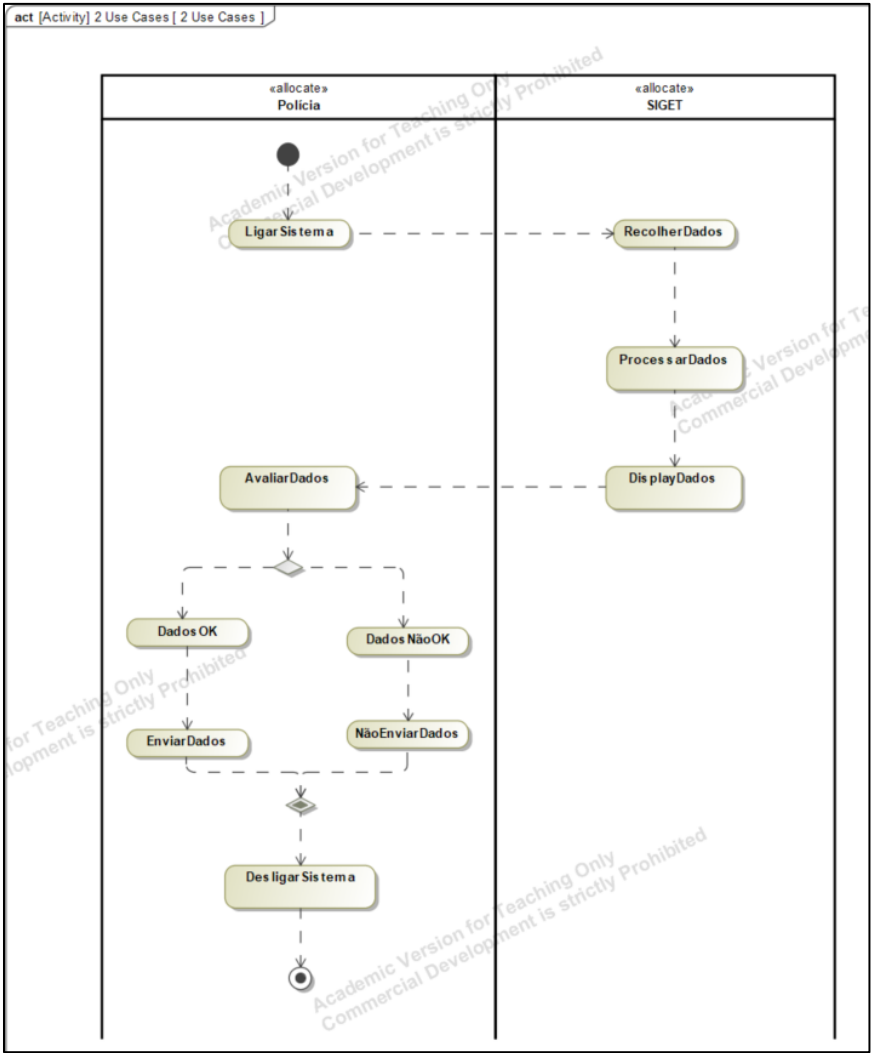


Imagem 4 - Activity diagram

De seguida construímos um Activity Diagram para fazer o Function Analysis das restantes partes do sistema.

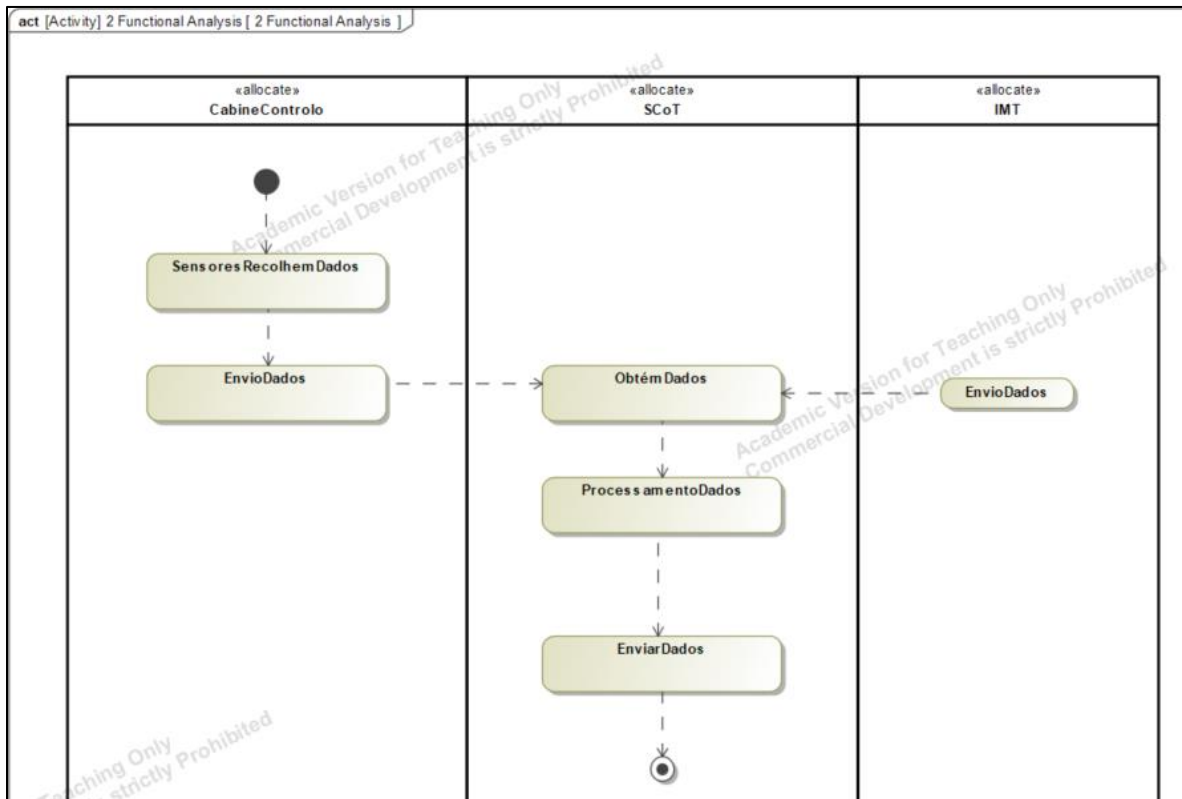


Imagem 5 - Function Analysis

Considerando que já temos uma ideia, ainda que um pouco rudimentar, sobre o funcionamento do sistema é necessário dar início ao esquema da sua montagem física, criando um Internal Block Diagram para dar resposta às caixas presentes na coluna da estrutura.

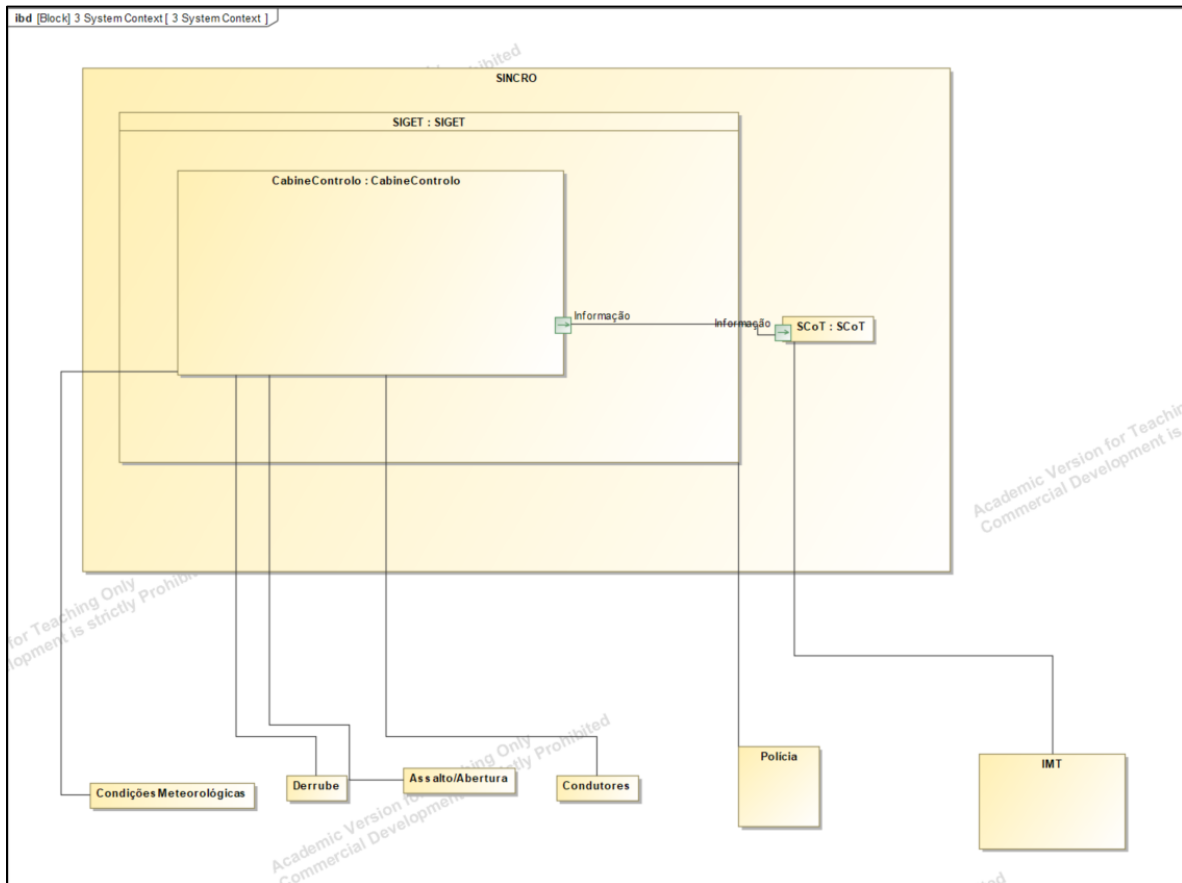


Imagem 6 - Internal Block diagram estrutural

Assim temos a estrutura do sistema e os seus intervenientes.

Com isto fazemos o mesmo para a Cabine de Controlo, que é um subsistema importantíssimo pois é a parte que obtém as informações e faz o processamento inicial dessa informação obtida no local desejado.

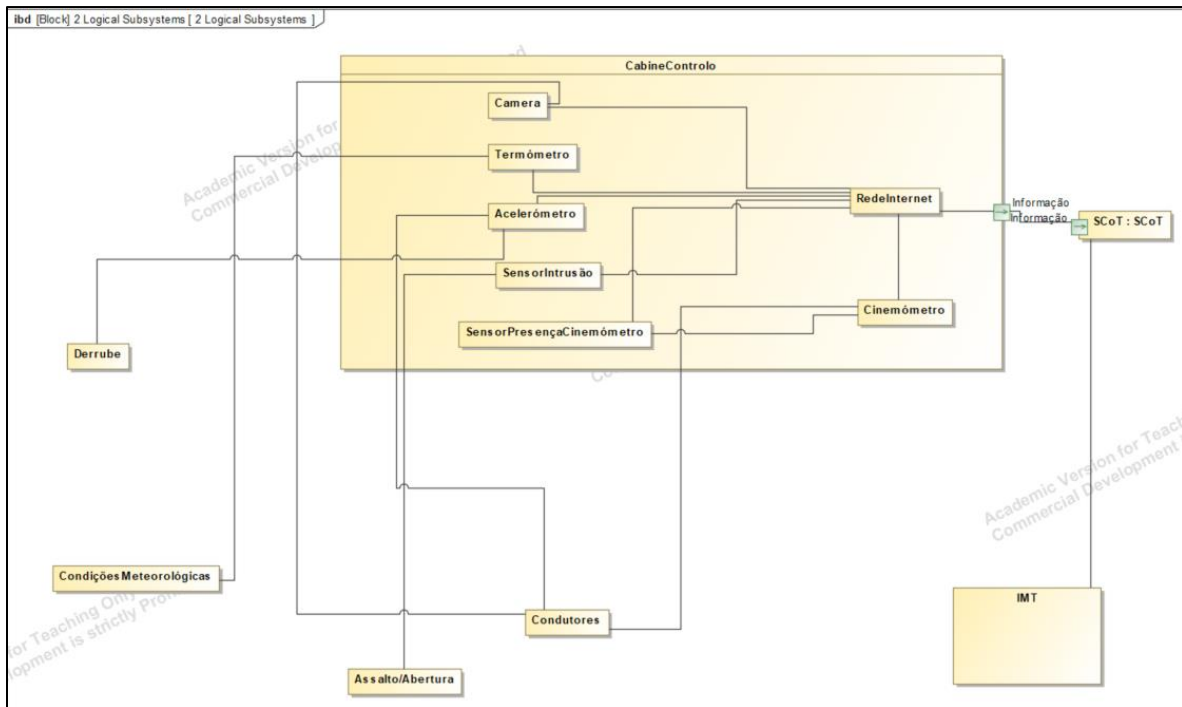


Imagem 7 - Internal Block diagram Cabine de Controlo

Assim já sabemos quais os componentes necessários para o funcionamento da Cabine de Controlo e temos uma ideia de como estão interligados.

O próximo passo é saber quais as informações que trocam entre si e para o exterior. Para isso, construímos um Internal Block Diagram recorrendo ao uso de Flow Ports para ilustrar o fluxo de informação.

Neste diagrama anterior entrámos no campo da solução em termos da estrutura do sistema, em seguida é necessário perceber o seu comportamento, respondendo à caixa de System Behavior e, para isso, procedemos à criação de um State Machine diagram.

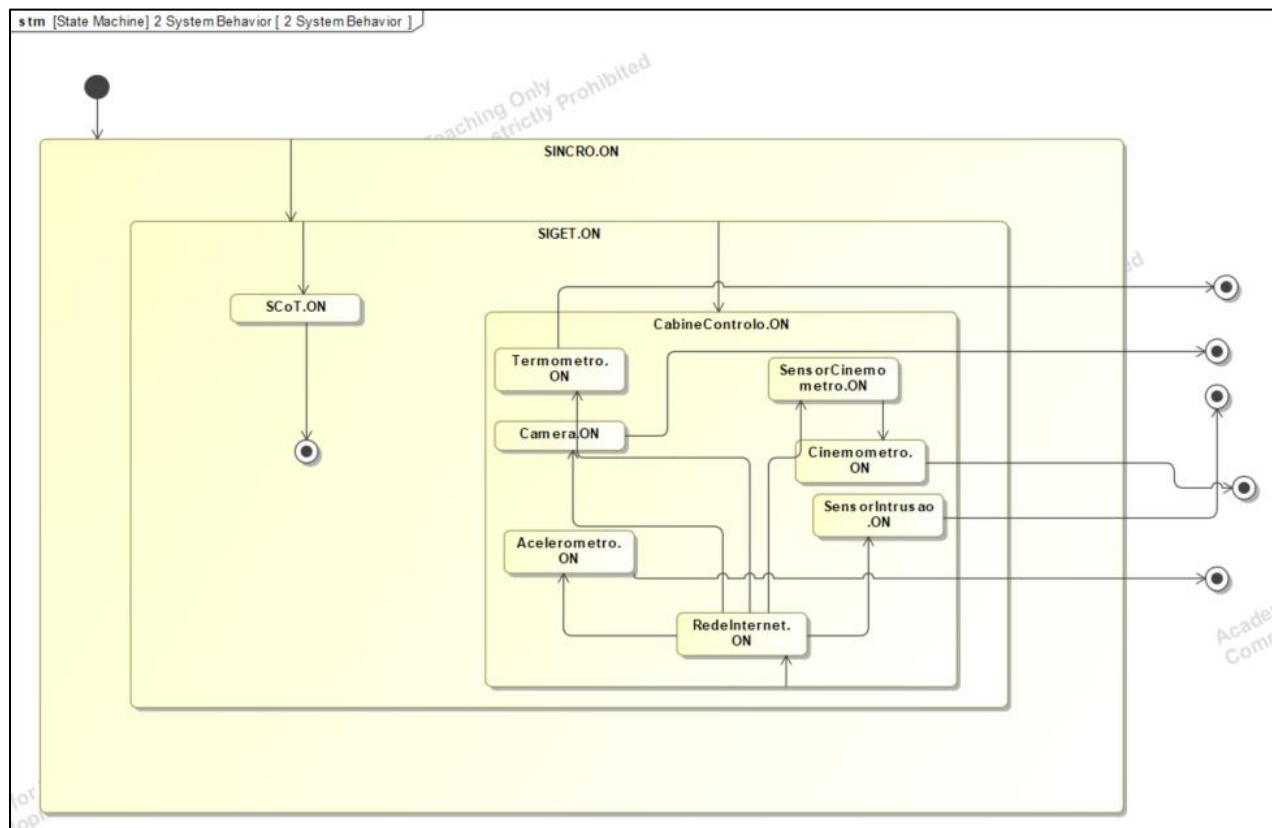


Imagem 9 - State Machine diagram comportamento do sistema

Com este diagrama sabemos quais as partes do sistema que levam ao funcionamento de outras, mas de uma forma pouco específica, então criámos um Activity Diagram com todas as interações e tipos de informação partilhados entre os vários componentes.

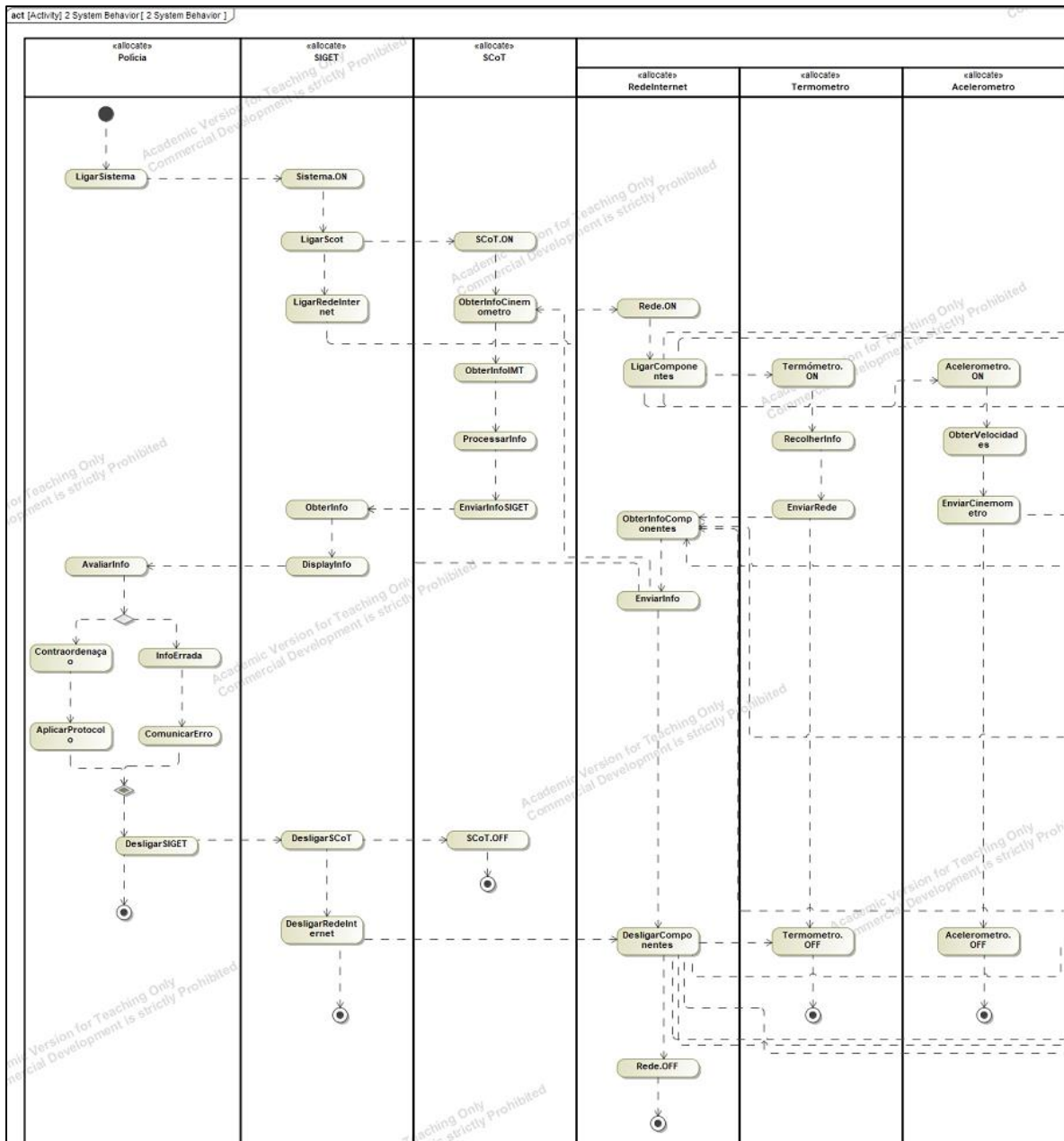


Imagem 10 - Activity diagram parte 1

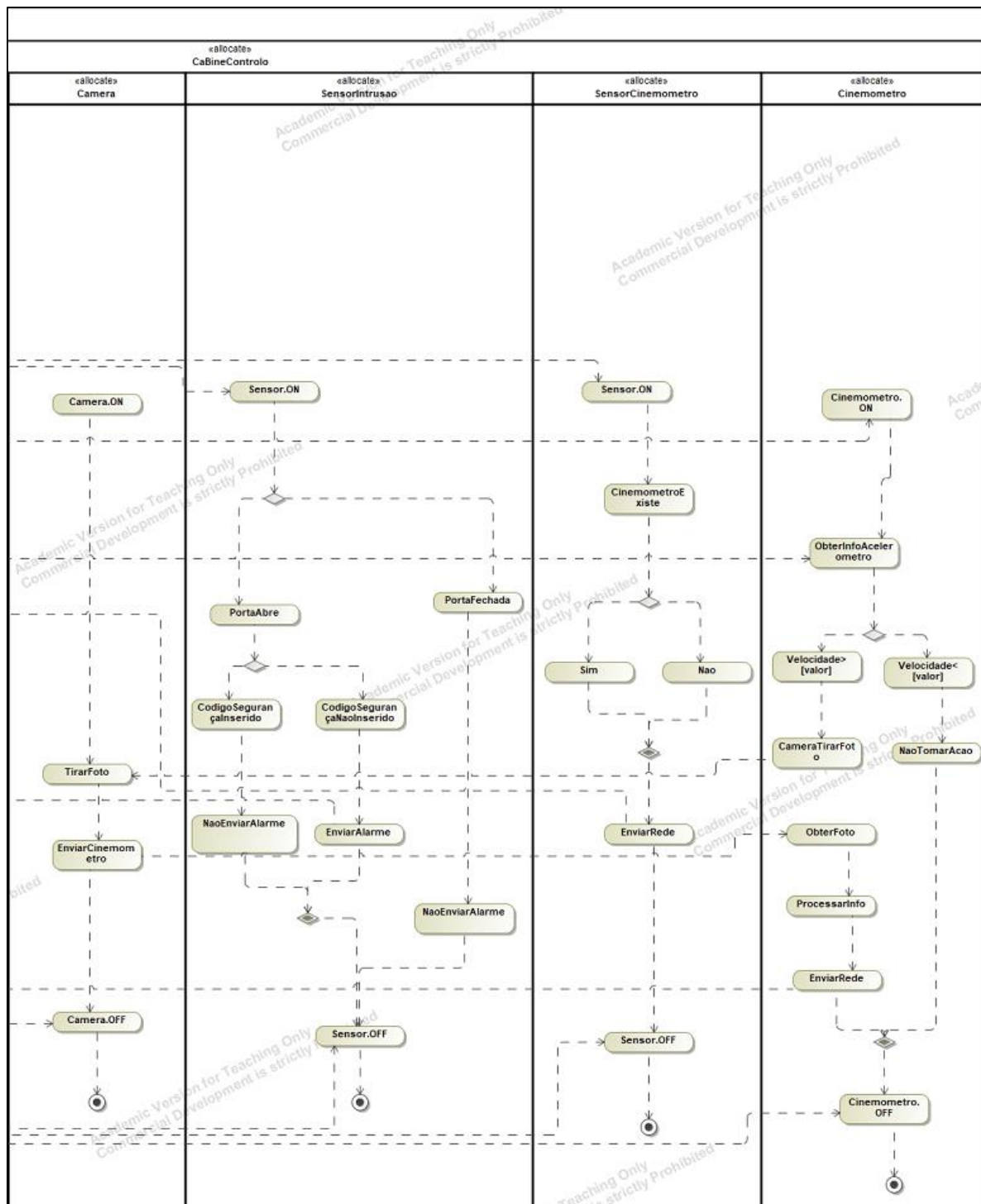


Imagem 11 - Activity diagram parte 2

Com este diagrama ficámos a saber como todo o sistema e subsistemas interagem e assim concluímos o nosso projecto deixando a implementação e as medições de eficiência para os especialistas destas áreas.

3. Conclusão

A modelação em SysML neste projeto revelou-se essencial para uma padronização das informações transmitidas entre todos os intervenientes desta rede SINCRO, quer engenheiros informáticos, quer entre a polícia, até ao funcionário da manutenção. Não é novidade que facilita a percepção do funcionamento dos sistemas, mas em comparação com o primeiro projeto proposto este revela-se mais completo, com funções e dependências mais abrangentes.

Foi nos possível também perceber a complexidade da rede e a sua relação com todas as interfaces por forma a contribuir para uma condução mais segura, sem acidentes.

4. Bibliografia

- [1] - <https://www.nomagic.com/support/quick-reference-guides> (consultado em 20/11/2018)
- [2] - Enunciado relativo ao projeto fornecido pelo professor Luís Osório
- [3] – ALEKSANDRAVICIENE, Aiste; MORKEVICIUS, Aurelijus. MagicGrid Book of Knowledge A Practical Guide to Systems Modeling using MagicGrid from No Magic. VITAE LITERA; Kaunas, 2018.