

**UNIVERSIDADE NORTE DO PARANÁ**

Sistema de Ensino Presencial Conectado

Superior em Tecnologia em Análise e desenvolvimento de sistemas

nome: Maykon Silva regis de araujo

**Sistema Zer@Dengue**

Arcoverde-PE

2020

nome: Maykon silva regis de araujo

**Sistema Zer@Dengue**

Trabalho interdisciplinar apresentado à Universidade Norte da Paraná (Unopar), como requisito parcial para a obtenção de média semestral na disciplina de:

Engenharia e Projeto de Software; Projeto Orientado à Objetos; Programação para WEB II.

Orientadores: Tutor Eduardo Viana de Almeida

Professora: Iolanda Cláudia Sanches Catarino;

Professor: Gilberto Fernandes Junior;

Professora: Vanessa Matias Leite.

Arcoverde-PE

2020

# Sumário

[1:INTRODUÇÃO 3](#_Toc40203578)

[2:DESENVOLVIMENTO 4](#_Toc40203579)

[**Tarefa-1: Engenharia e projeto de software** 5](#_Toc40203580)

1.2-Descrição do processo de desenvolvimento de software adotado.........................6

1.3-Método do desenvolvimento de software adota.....................................................7

1.4-Modelagem de atividade.........................................................................................8

1.4.2-Diagrama de classe...........................................................................................10

1.4.3-Diagrama de atividade.......................................................................................12

1.4.4-Diagrama de sequencia....................................................................................14

1.5-Diagrama de entidade e relacionamento.............................................................16

1.5.1-Estrutura logica e fisica.....................................................................................18

[**Tarefa 2: Projeto Orientado à Objetos** 19](#_Toc40203581)

[**Tarefa 3: Programação para WEB II** 20](#_Toc40203582)

3.1-Documentação do sistema em funcionamento.....................................................24

CONCLUSÃO**.............................................................................................................**25REFERÊNCIASBIBLIOGRÁFICAS**...........................................................................26**

# INTRODUÇÃO

O tema proposto para esse trabalho e muito interessante pelo fato de estamos vivendo uma situação real que e o mosquito da dengue( Aedes aegypti), e um problema que nós temos que combater todos os dias, e como combater esse problema da sociedade, já temos métodos a serem adotados para a não proliferação do mosquito, mais como cidadão de bem posso contribuir a mais com a sociedade, foi ai que surgiu a ideia de desenvolver um sistema que ao fazer o cadastro você passa a denunciar e monitorar casos onde possa haver foco do Aedes aegypti enviando fotos consultas áreas já mapiadas e muito mais e ao fazer a denúncia, haverá equipes próprias das prefeituras a ser deslocadas para o local do ocorrido para se fazer a limpeza do local. Mais o mais importante que fica nesse aprendizado e todos fazerem sua parte que não haverá mosquito.

# DESENVOLVIMENTO

Nesse trabalho foi proposto a desenvolver um software que a qualquer pessoa fazer um cadastro de todos os dados ela possa monitorar e denunciar locais com foco da dengue, e a parti de sua denúncia equipes da prefeitura possa ir aos locais fazer a devida limpeza, com a ajuda das disciplinas de engenharia e projeto de software, projeto orientado a objeto, e programação para web 2 foi pedido para desenvolvemos um sistema capaz de ao criar uma conta e fazer um descritivo do problema, indicando o endereço da ocorrência, inclusive podendo enviar fotos, sendo que, posteriormente o usuário receberá um relato da ação tomada pelo Ministério da Saúde. Na primeira tarefa fazer um plano de gerenciamento de projeto de software, a descrição do Processo de Desenvolvimento de Software a ser adotado, a modelagem das atividades básicas de Análise de Sistemas e Projeto de Sistemas, já na tarefa 2 foi pedido fazer uma breve pesquisa e indique o tipo de linguagem orientada a objetos mais adequada ao projeto, defini frameworks e sua importância em projetos de sistemas orientados à objetos. Na última tarefa 3, desenvolver o portal do sistema. Para isso, você deverá implementar um site que possui uma página para cadastro do usuário, os campos com todos os campos necessários para um cadastro e após o cadastro realizado, o usuário deve ser redirecionado para a página da denúncia. Nesta página, deverá conter os seguintes campos: Endereço da ocorrência, Campo para postagem de foto mostrando o local com o foco de dengue, Descrição obrigatória com o relato do local. Toda essa parte de cadastro ser desenvolvido com as linguagens php html e css.

## **TAREFA-1: Engenharia e projeto de software**

**1-**Bom recebemos o pedido do cliente avaliamos em reunião, é coletamos os requisitos definimos, foi elaborado um prazo de entrega, também foi definido o orçamento e também serão feitos interações semanais com o cliente no que chamamos o jogo do planejamento que e nos desenvolvedores e cliente reúnem-se para priorizar as funcionalidades. e começamos a montar. Nosso plano de gerenciamento foi elaborado usando os seguintes métodos, optamos por usar um dos processos ágeis de desenvolvimento que o Extreme Programming (XP) que aborda um dos fatores que e crucial para o desenvolvimento que o de feedback e muitos outros então, nós montamos uma equipe em que todos os participantes terão sua função e começamos a definir o seu papel para cada um, é entender o que o cliente propôs, optamos por usar o xp pelo fato de se projetar a equipe(não que outros métodos não tenham equipes) de cada um possa ter sua função pôr o método se apoiar em quatro pilares que são o feedback, comunicação a simplicidade e coragem, então introduzimos algumas técnicas de levantamento de requisitos, usaremos também as próprias técnicas do xp para que possamos entender o que o cliente que, é o que ele deseja.

1.2- Descrição do Processo de Desenvolvimento de Software adotado

Bom o método adotado foi o Extreme Programming (xp), começamos por fazendo uma reunião e também ao longo do processo e reuniões semanais e seguindo todas as práticas da xp com o cliente que chamamos de jogo de planejamento(Planning Game) para entender o que está sendo proposto então começamos usando algumas técnicas de levantamento de requisitos, usamos a entrevista é por sequencia uma análise de requisitos e uma aplicação de questionário, coletamos todas as informações e começamos a trabalhar. Começamos então a desenvolver pequenas versões do projeto auxilia muito no processo de aceitação por parte do cliente, a análise de tudo o que foi proposto foi definido também os requisitos funcionais e não funcionais do software e elaborado um prazo de entrega e também o custo do software então começamos a definir toda a equipe de trabalho cada um com sua função seguindo o método da xp. Seguindo adiante o Projetos Simples, Testes de Aceitação que e o cliente vai nos visitar para conhecer o desenvolvimento junto das reuniões que são testes construídos pelo cliente e conjunto de analistas e testadores, para aceitar um determinado requisito do sistema, também reuniões em pé, para não se perder o foco nos assuntos, produzindo reuniões rápidas, programação em pares que é a programação em par/dupla num único computador. Geralmente a dupla é formada por um iniciante na linguagem e outra pessoa funcionando como um instrutor. E seguindo todos os outros passos da xp.

1.3-Método (s) de Desenvolvimento de Software adotado

Nessa parte do desenvolvimento nos optamos por introduzi também a Unified Modeling Language (UML) no processo de desenvolvimento e modelagem por ter diagramas que iram facilitar o entendimento do cliente no projeto, então nós usaremos o diagrama de use cases o diagrama de objeto e o diagrama de atividades e também por ser conhecida mundialmente e todos que trabalham nessa área de ti de desenvolvimento terem familiaridade com a UML, a ferramenta case usada será a astah.

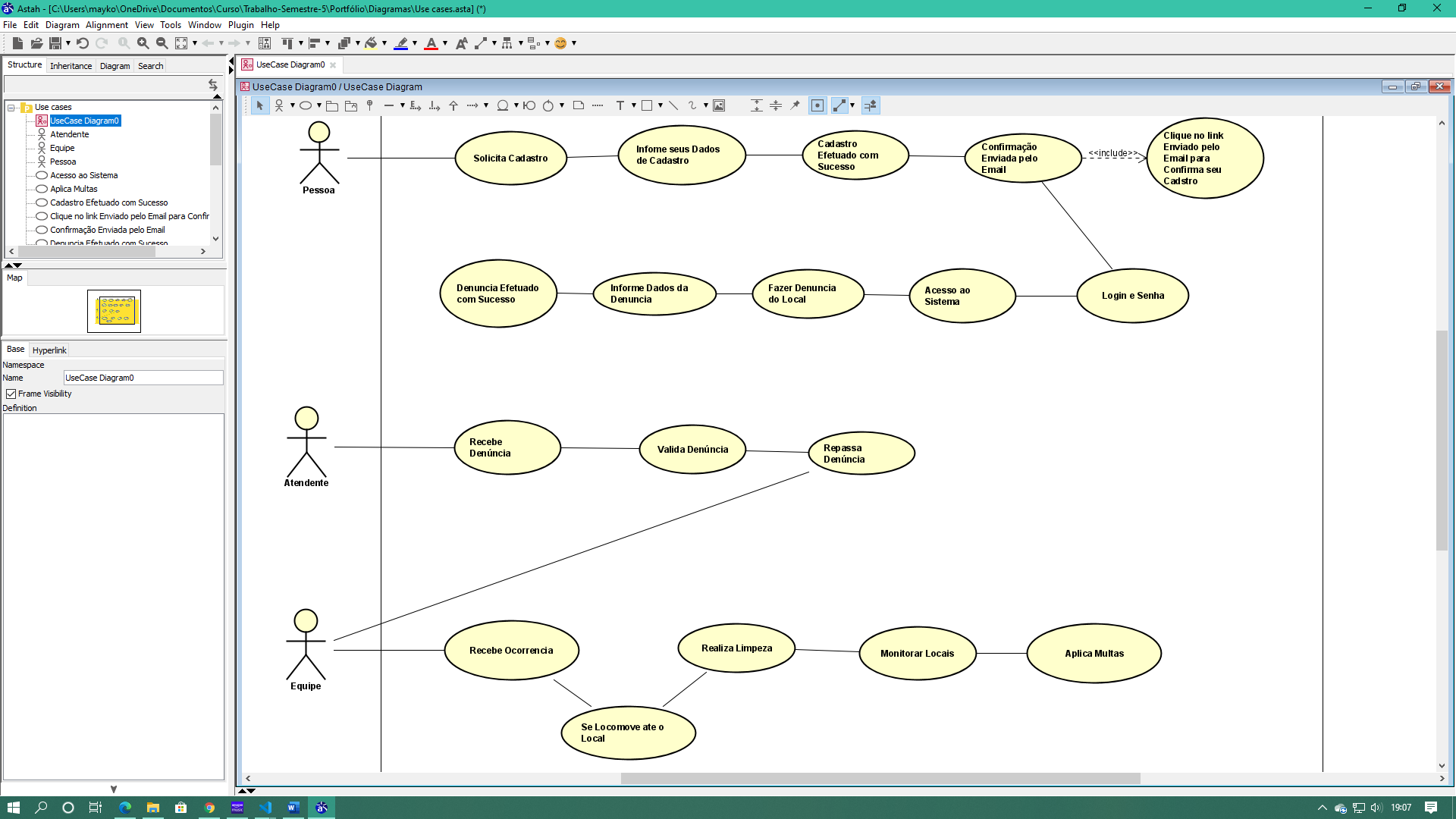
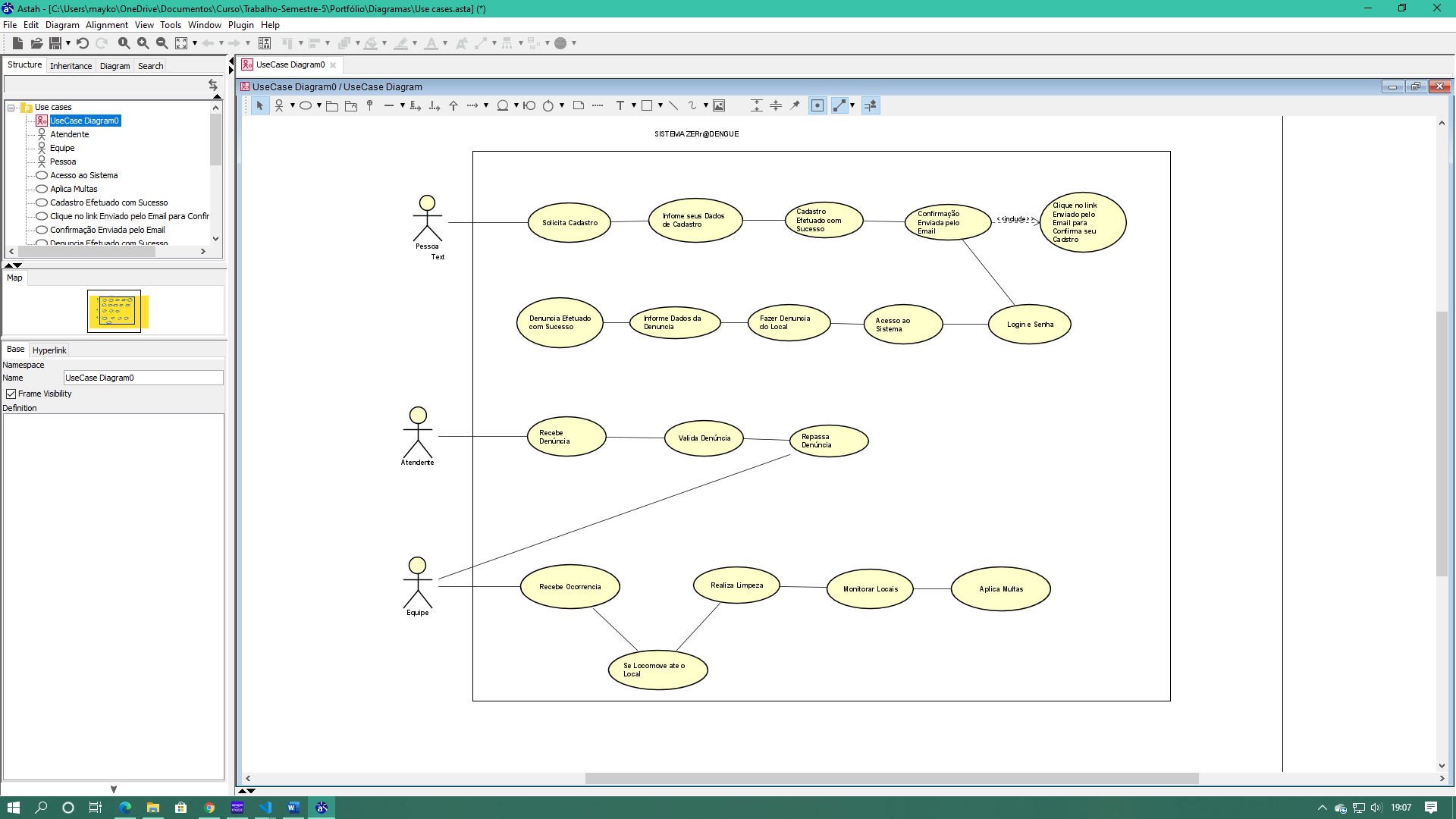
O diagrama de use cases foi usado nesse projeto para retratar o funcionamento do sistema para o cliente para que ele entenda como o sistema se comportara e para visualização das principais funções que são o que o software irá fazer no dia a dia sem se preocupar com a programação como os dados serão tratados nem coisa desse tipo.

Já no diagrama de classe mostrar a relação dos envolvidos no software e mostrar também as classe e seus relacionamentos, mostrar também os dados que serão pedidos para o cadastro, esse diagrama tem como objetivo mostrar a estrutura do sistema como ela irá se comporta, seus atributos e métodos que são o que o sistema tem e o que ele faz.

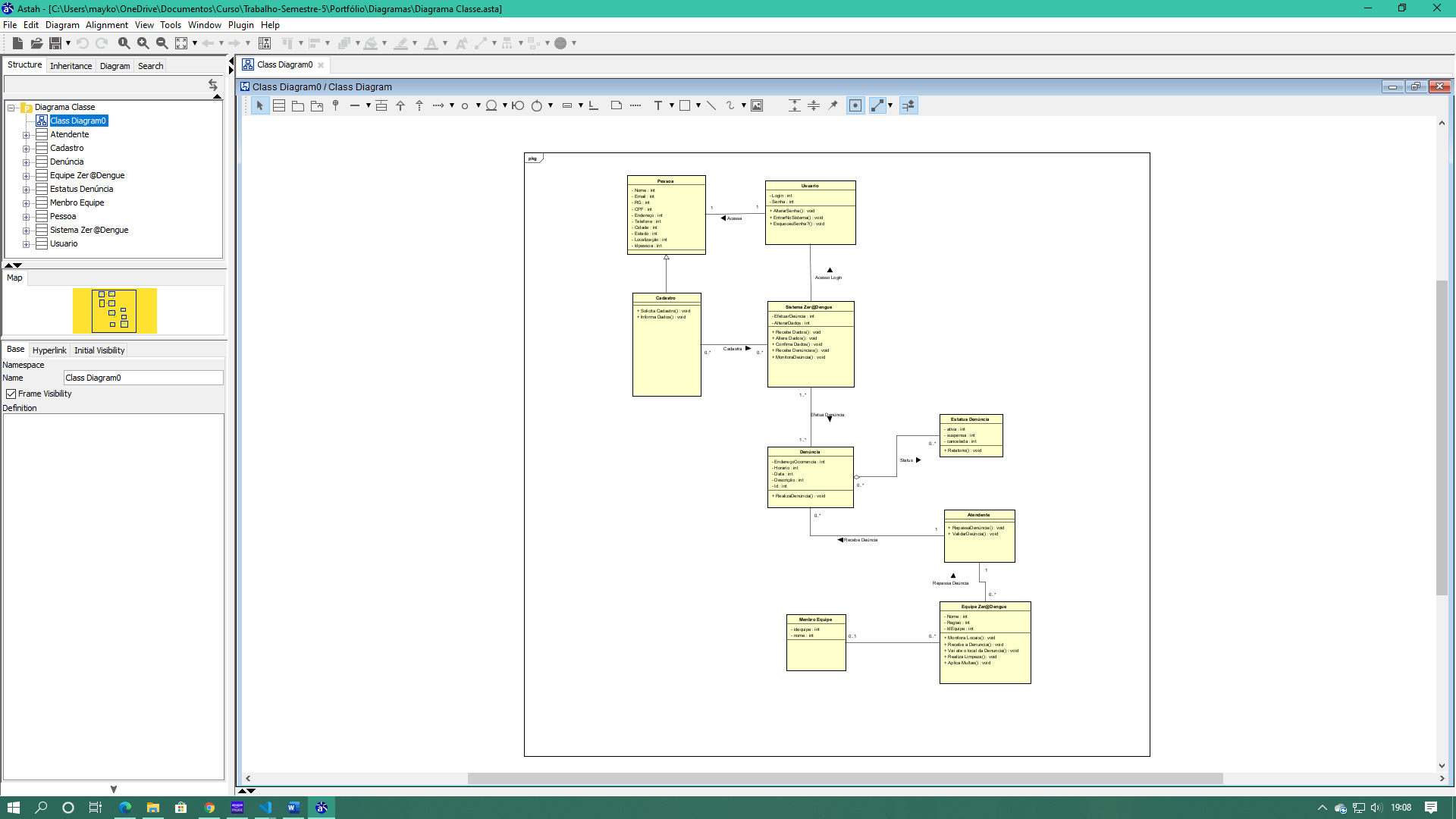
O diagrama de atividades ele tem como uma extensão dos fluxograma ,mostra o fluxo de uma atividade para outro que e todo o passo do sistema desde o cadastro até a parte da denúncia que o solicitante irá usar, nesse diagrama representa o funcionamento do sistema, todo o passo a passo desde a solicitação até a parti final do cadastro e da realização da denúncia.

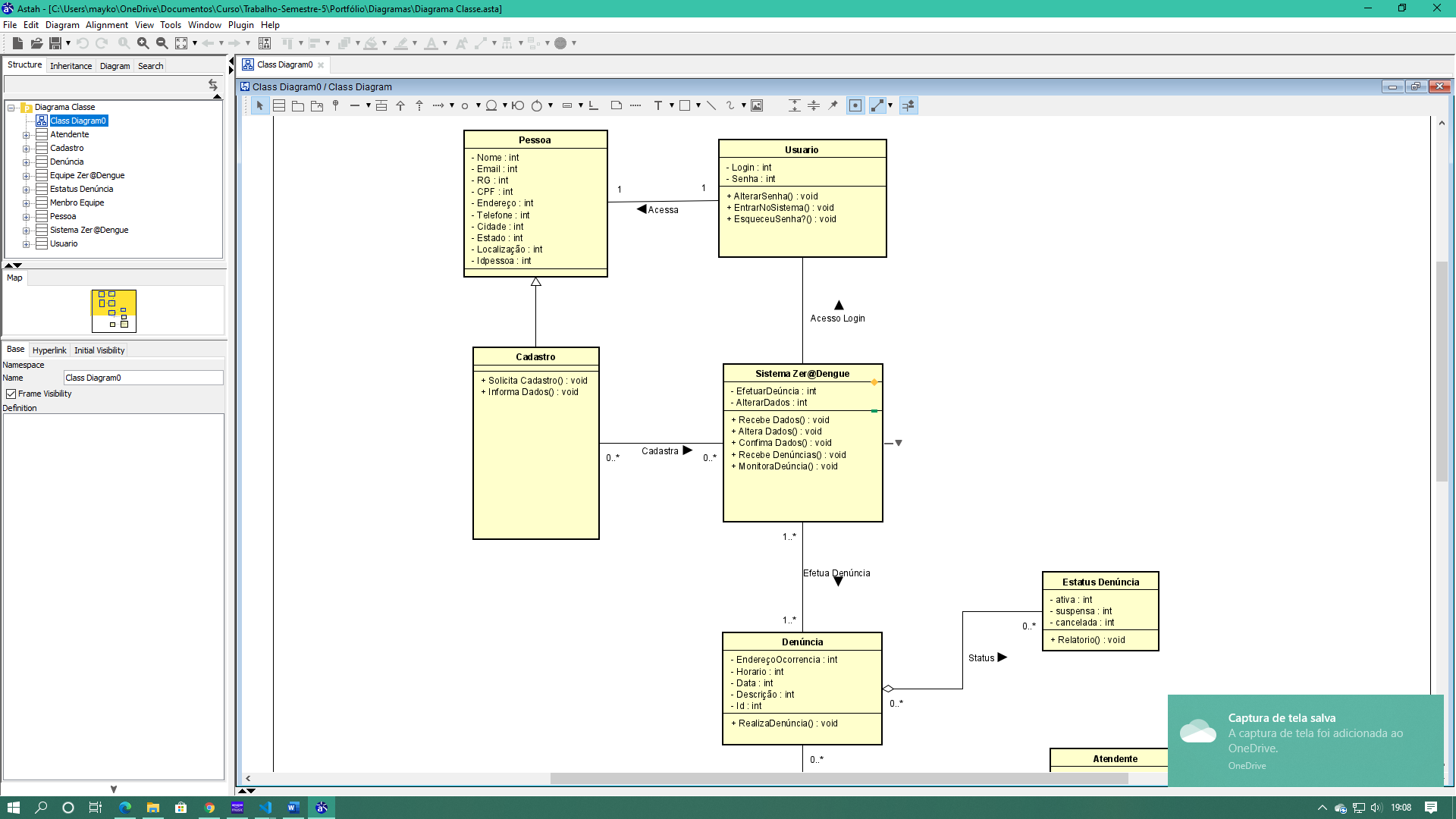
O diagrama de sequência se baseia no diagrama de casos de uso, esse diagrama tem como proposito mostrar as mensagens que o ator enviara para um objeto no caso o sistema com o objetivo de executar uma ação, mostrar toda a sequência das interações do ator com o sistema, aqui teremos dois diagrama o do ator pessoa interagindo com o software e o da equipe validando os dados recebidos após o ator realizar a denúncia.

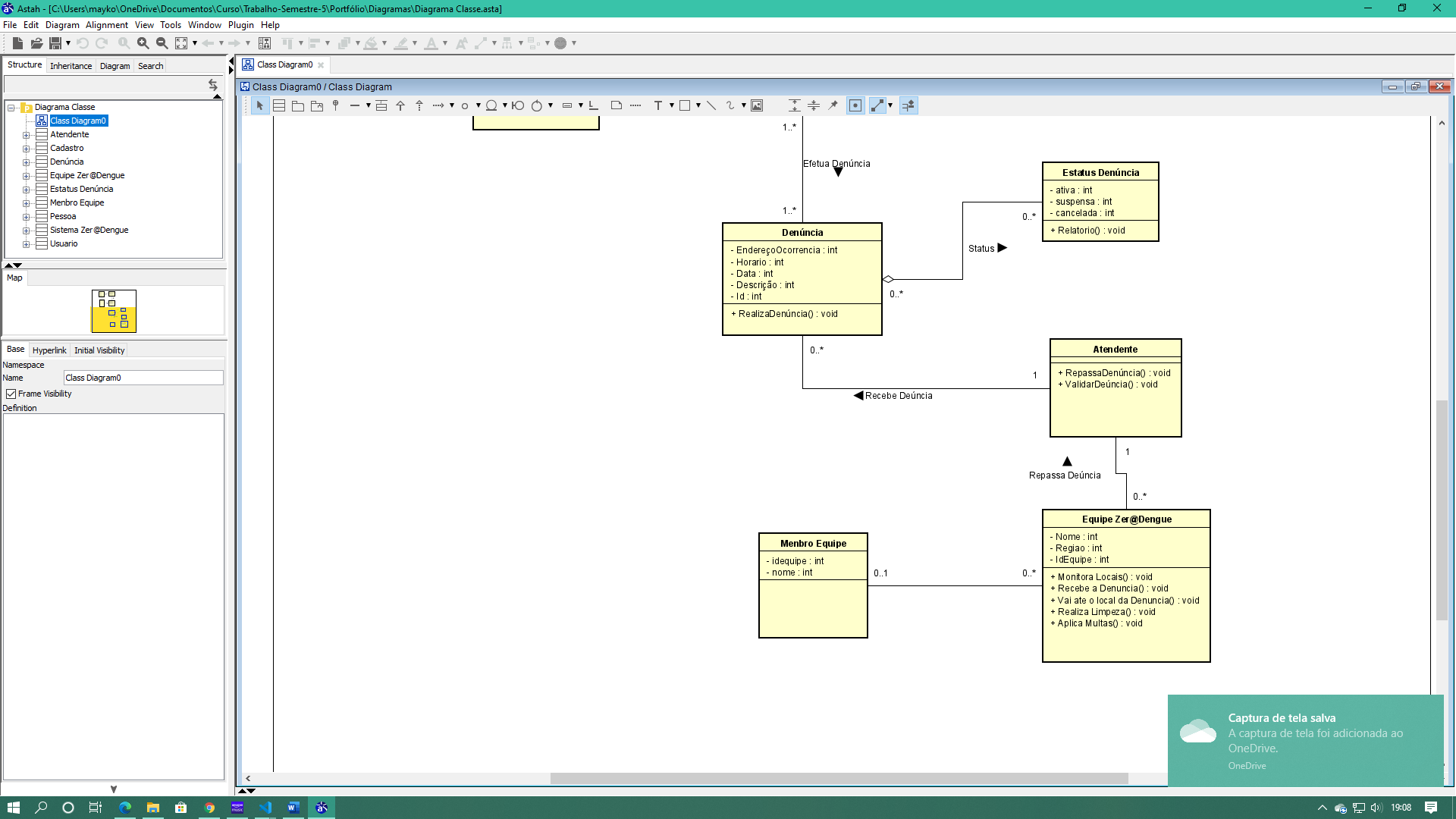
**1.4**-**Modelagem de atividades Diagramas**

1.4.1-DIAGRAMA DE USE CASES

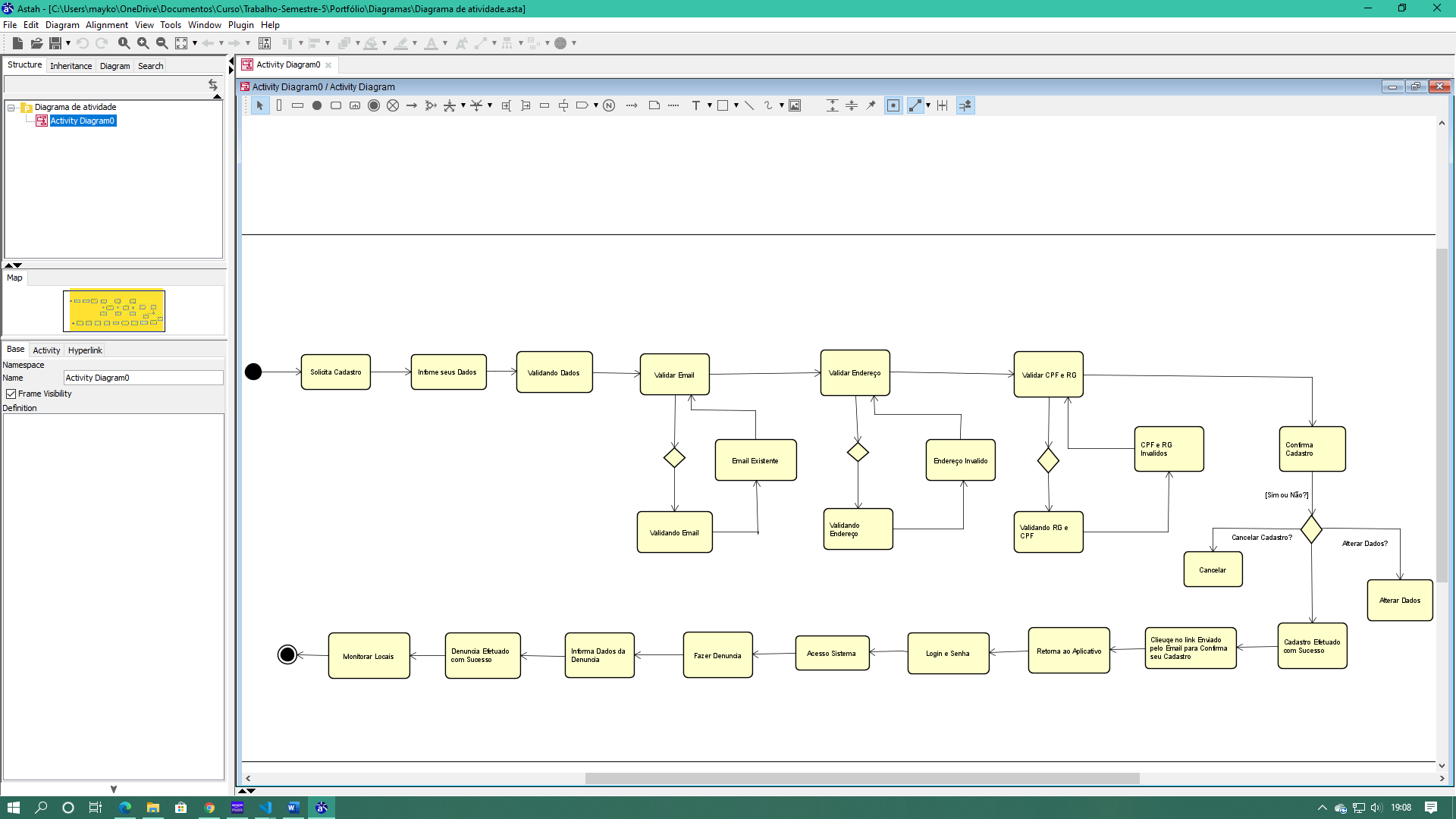
Bom nesse diagrama de use cases procurei comecei procurando entender como funcionaria esse sistema que nos foi proposto a ser desenvolvido, primeiro caso identificar os atores do sistema eu identifiquei dois atores a pessoa e a equipe que vai receber a denúncia e monitorar os locais e tudo mais, também procurei definir os requisitos funcionais e não funcionais e os requisitos do sistema, bom partindo desse ponto a pessoa solicita o cadastro aparecera uma tela para ela preencher com os dados dela depois ela irar receber uma mensagem de cadastro efetuado com sucesso logo em seguida será enviado um link para o e-mail da solicitante para confirmação e só será liberado o sistema quando ela clicar no link, que serve para confirmação real da pessoa, depois disso ela estará livre para realizar a sua denúncia, informando os dados da denúncia e posteriormente monitora aquele local. Na parte da equipe suas funções e receber a denúncia, validar a denúncia porque esses sistema podem o correr de engraçadinhos realizar cadastro fake e comprometer o sistema, depois a equipe ira até o local, realizara a denúncia, e posteriormente aplica multas em casos de local público for constatado pessoas jogando lixo, que causara na proliferação do mosquito da dengue. **1.4.2-DIAGRAMA DE CLASSE**

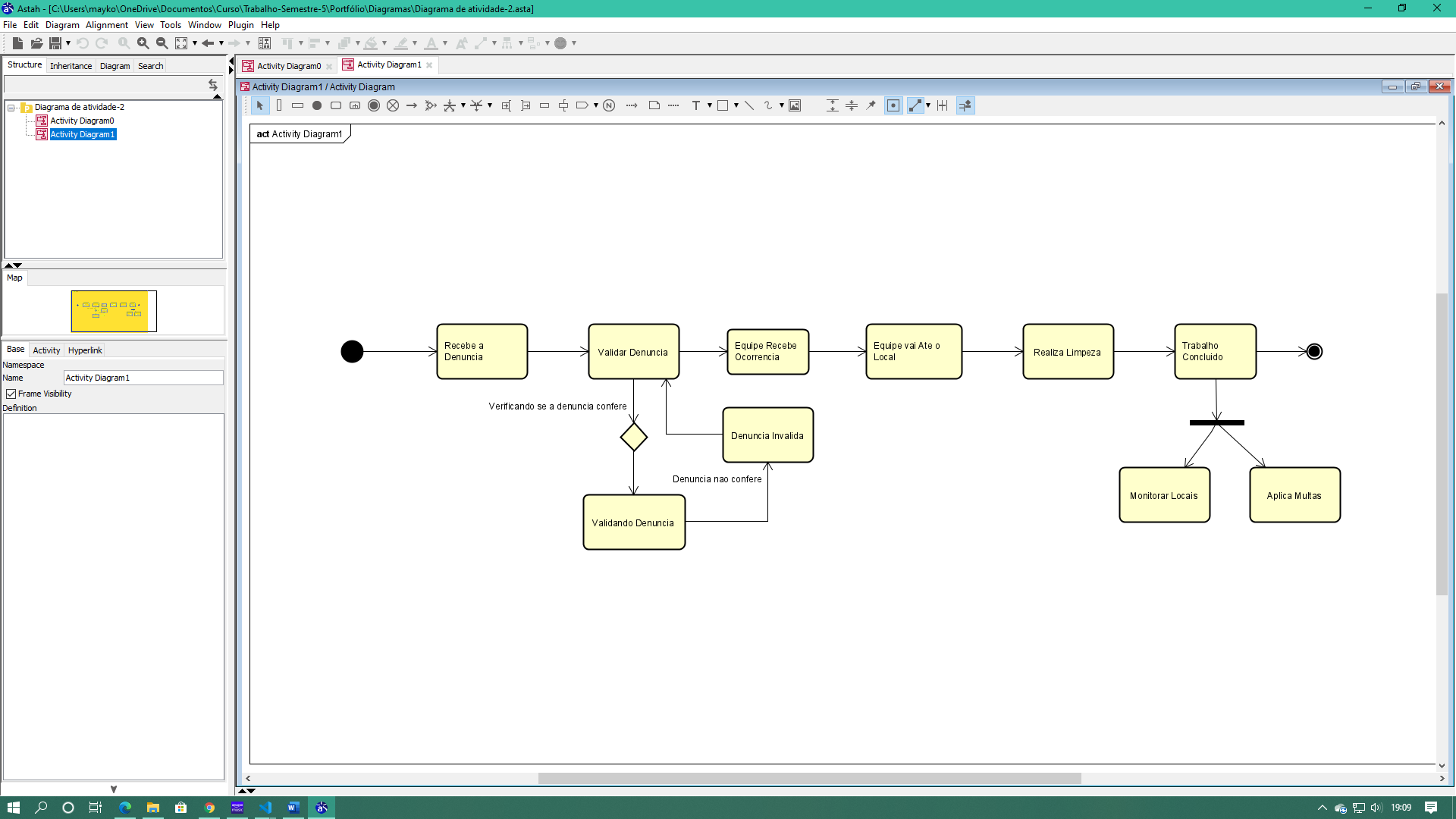
****

****

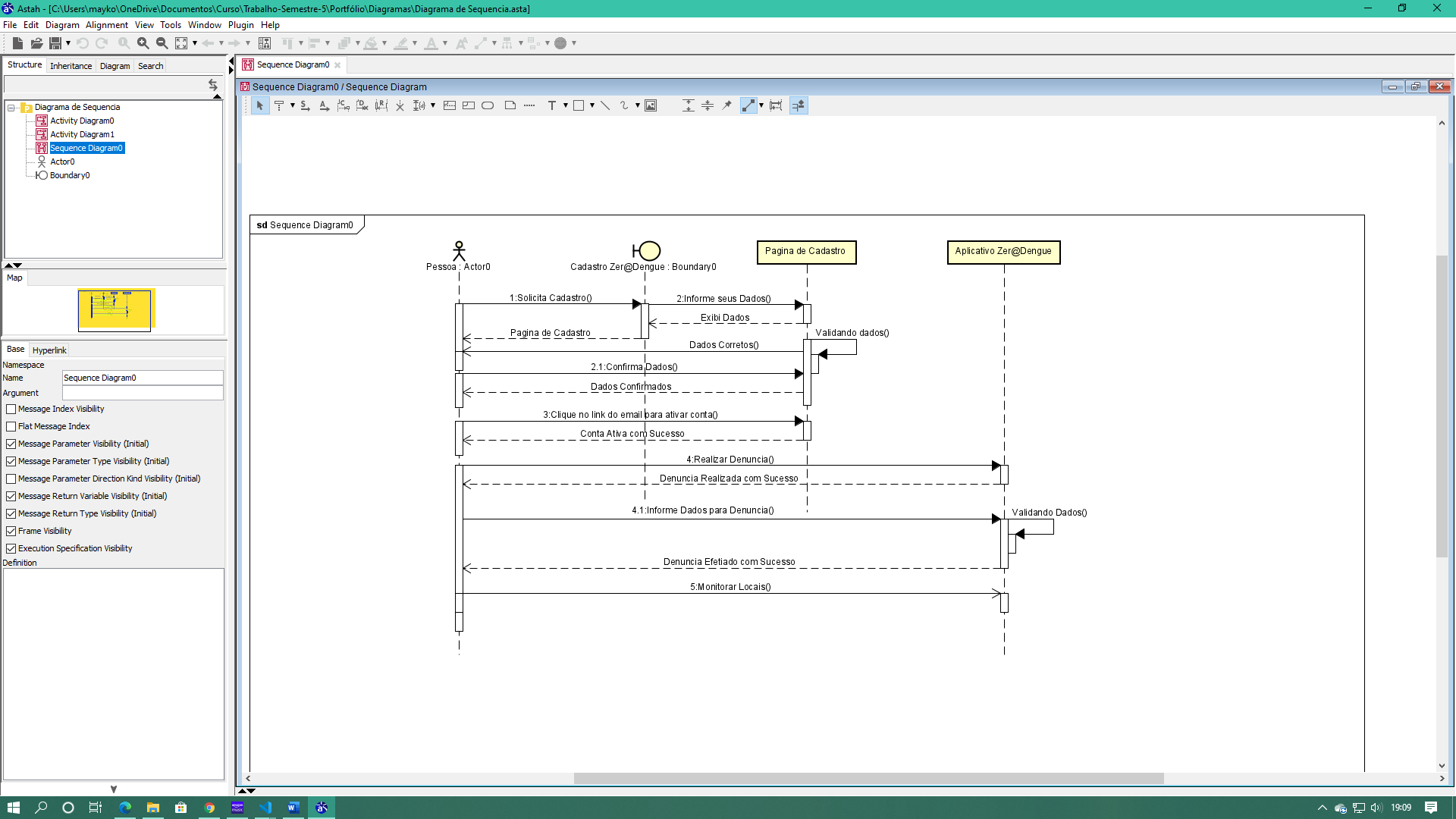
****

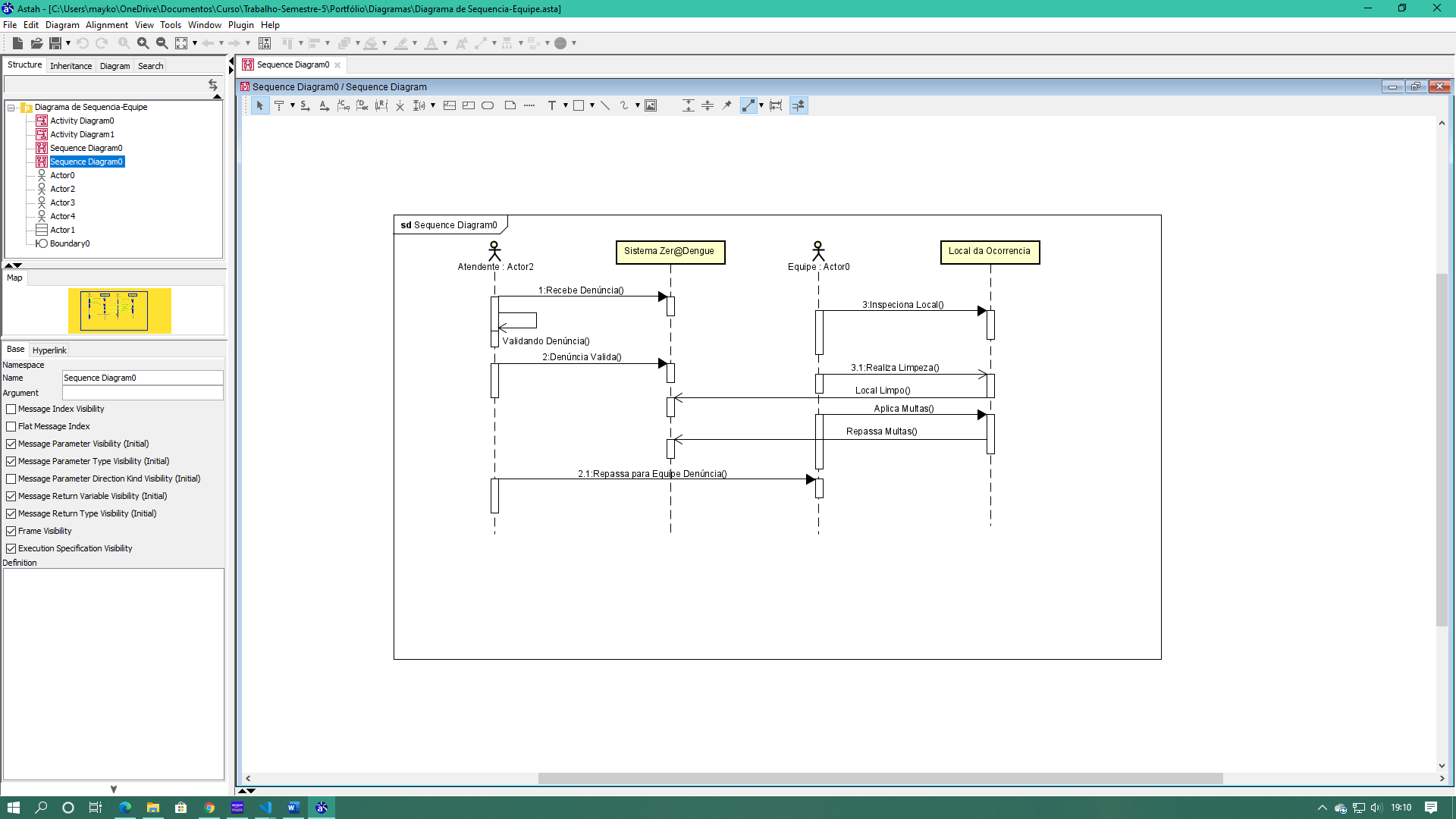
Já o diagrama de classe foi definido as classes principais do sistema, a interface que foi dado o nome de sistemaZer@Dengue que por onde a pessoa irá fazer todas as funções, também foram criados as classes pessoa, usuário, cadastro, que serão por onde ela irá realizar o cadastro e por sequencia realizara seu login, que dará acesso ao sistema zero dengue, onde irá ter a cesso a classe denúncia lá efetuara sua denúncia, logo em seguida da efetuação da denúncia a classe atendente que validara essa denúncia para pode repassar para a equipe ir até o local realizar a vistoria. Todas as classes foram definidos seus atributos e métodos de acordo com o que foi solicitado pelo contratante. Esse diagrama o objetivo dele e estrutura o sistema com suas classes, e como funcionara seus relacionamentos para o entendimento de toda a esquipe.

1.4.3-Diagrama de Atividade

****

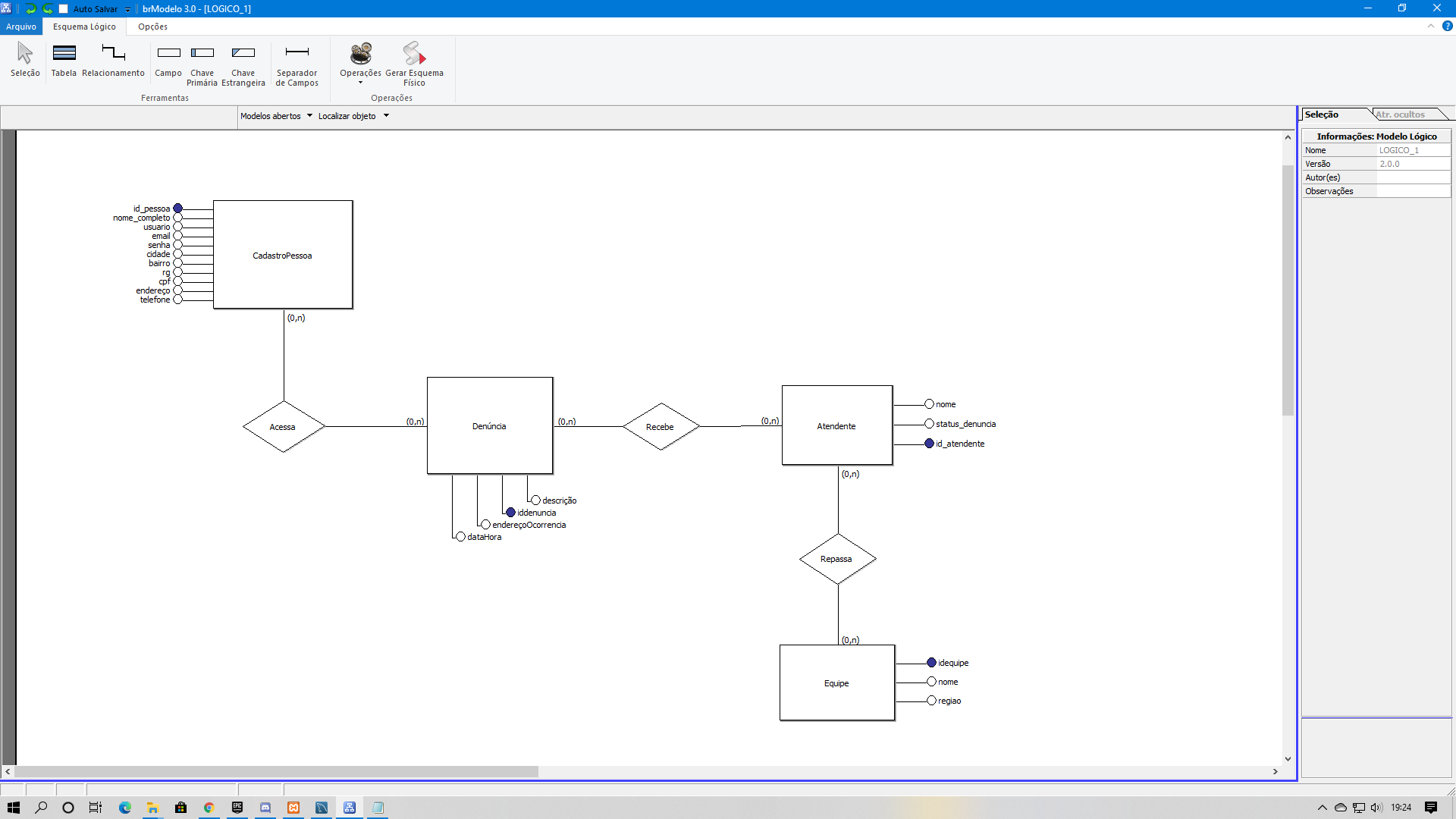
Nesse diagrama o foco dele e mostrar o passo a passo do sistema, no caso desse sistema seria mostrado o passo do cadastro pedido para dar acesso ao sistema, primeiro a pessoa informa seus dados, o software vai validar alguns para ter certeza que e uma pessoa, depois o sistema vai seguindo os passos validando os dados, não todos mais alguns necessários no final terá uma opção de alterar dados ou cancelar ou seguir em frente para confirma seu cadastro logo em seguida ela será redirecionada para a interface que lá poderá realizar a denúncia, novamente será pedidos os dados da denúncia, local do foco, uma descrição, localização e endereço, e em seguida monitorar sua denúncia no local. No segundo diagrama temos a participação da equipe que estará ligado ao sistema qual sua função, basicamente será receber a denúncia a central ou a logística por trais envia a equipe para realização da limpeza e monitoramento, será receber a solicitação valida se são corretas as informações faz a limpeza e depois monitorar para aplicar multas caso ocorra de pessoas estarem sujando o local novamente, porque trata de saúde pública todos devem fazer sua parte.

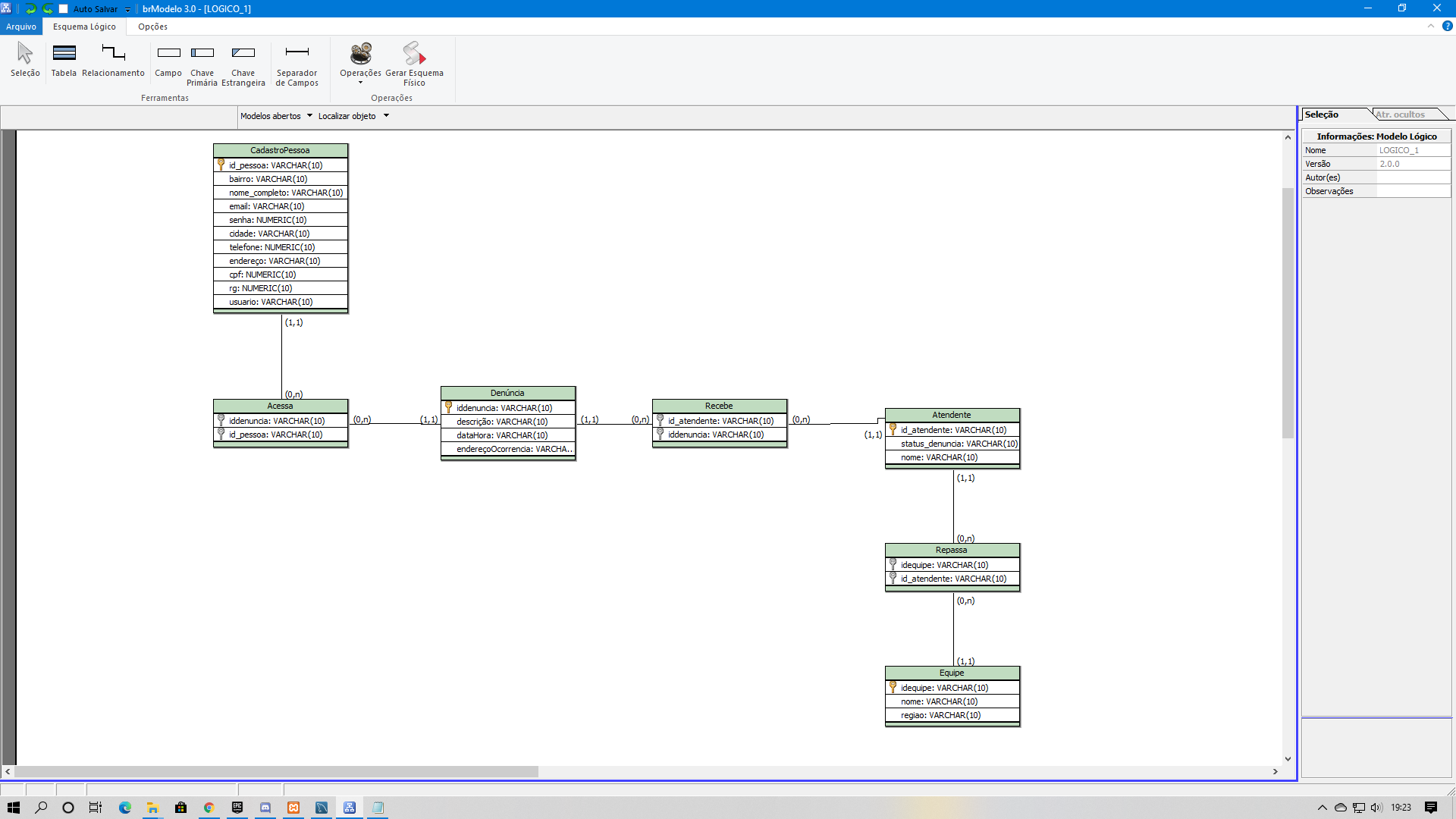
1.4.4-Diagrama de Sequencia

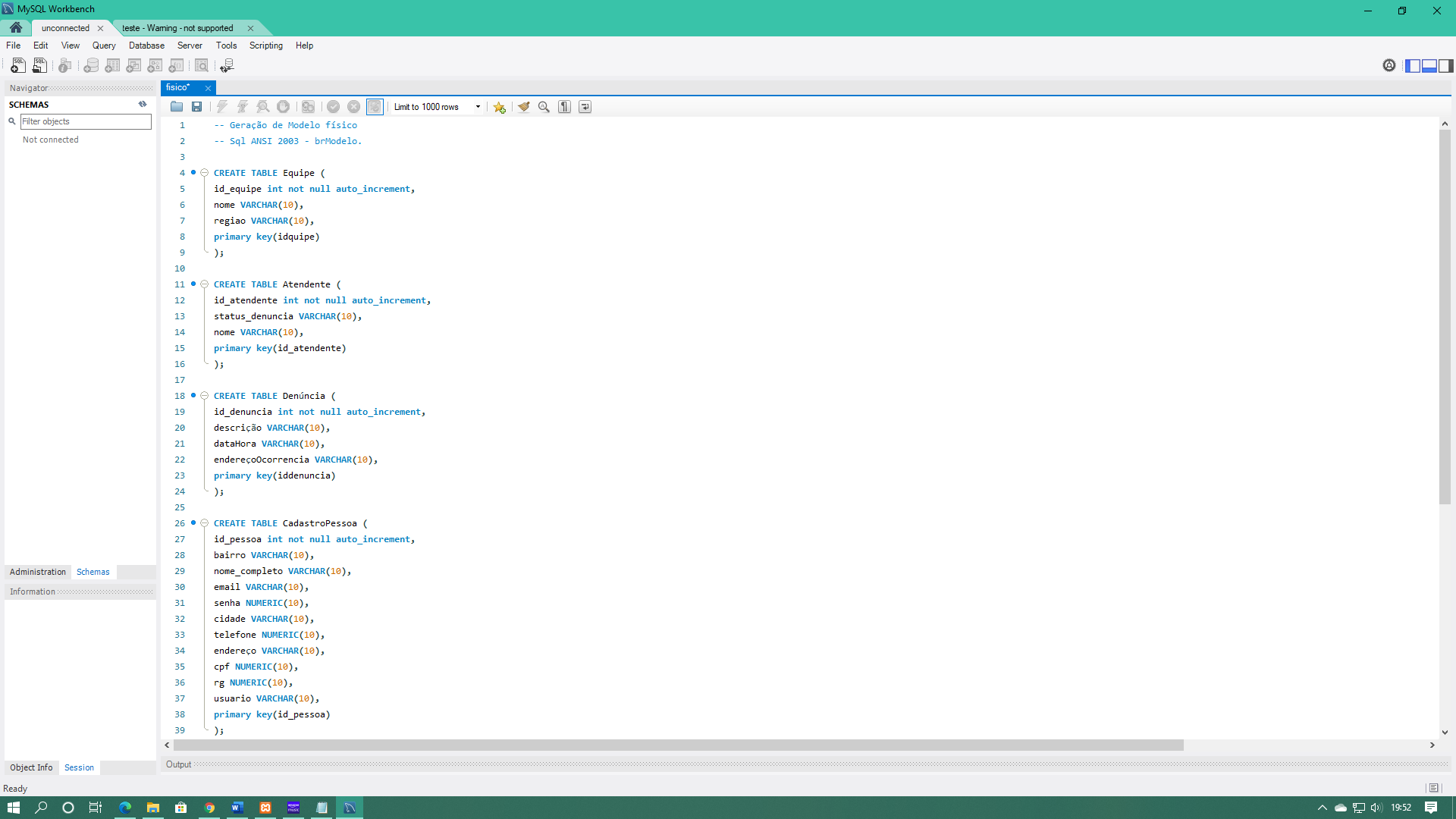
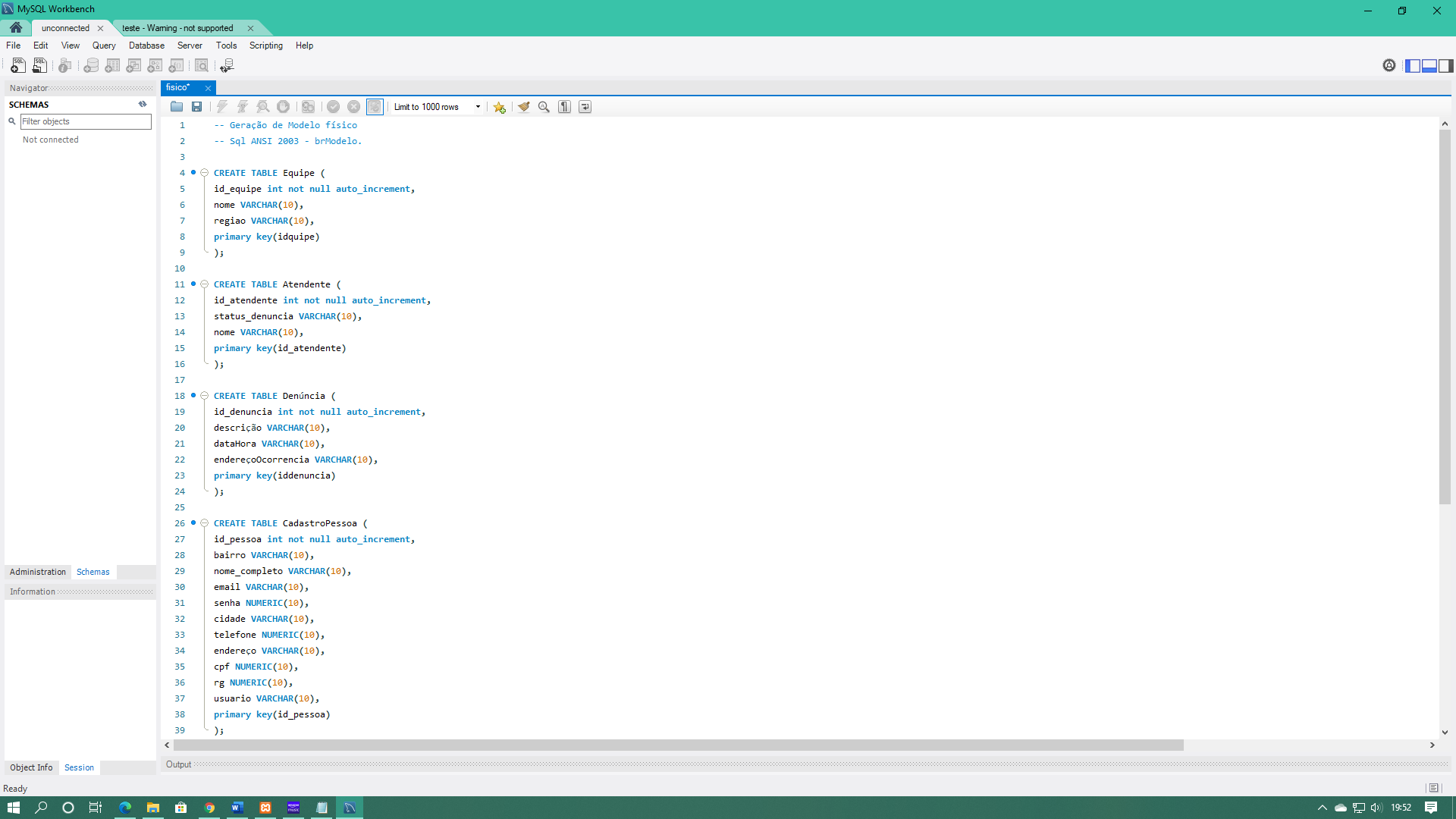
****

Este diagrama se baseia no diagrama de casos de uso, juntamente para exibir como será a interação dos atores com o sistema, mostrando os comandos(mensagens) que serão disparado ao solicitante solicitar o cadastro, retornando a ele a página e em seguida pede pra ele colocar seus dados válida esses dados se estiver tudo correto confirma seus dados, depois do cadastro efetuado será direcionado a página para efetuar sua denúncia. No diagrama da equipe será exibido sua função perante ao sistema, que são receber a denúncia, validar a denúncia e se locomover até o local para efetuar a limpeza ou aplicação de multa.

**1.5**-**Diagrama Entidade e Relacionamento**

.



****

1.5.1-Estruturas Lógicas e Físicas

Nessa arquitetura lógica tem como função mostrar como será o funcionamento do banco em si, já com os chaves e tipos primitivos definidos, nessa parte foram criadas 3 tabelas que se relacionam com o sistema. Já a arquitetura física já está definida a parte do código em si, ou seja, pronto para ser criado num SGBD escolhido no caso desse projeto foi escolhido o MySQL.

## **Tarefa 2: Projeto Orientado à Objetos**

2.1-Um tipo de linguagem que se adequaria nesse projeto seria PHP por ser uma linguagem conhecida fácil de entendimento, e ser orientada a objeto e permiti fácil ligação com banco de dados que nesse projeto será de importância utilização para receber os dados dos cadastrastes. Já o padrão utilizar seria o MVC.

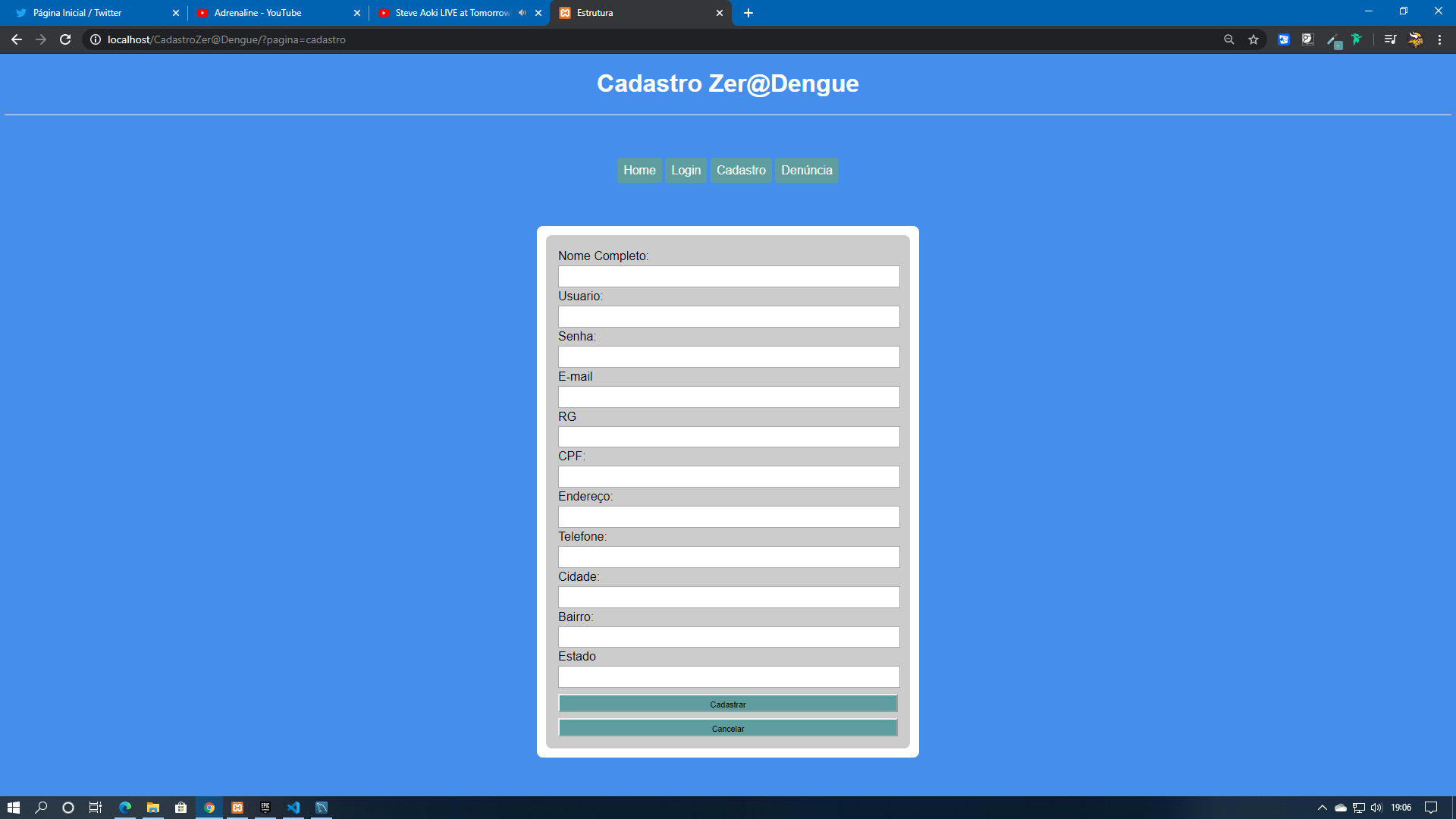
2.2- Assim um Framework tem como principal objetivo resolver problemas recorrentes com uma abordagem genérica, permitindo ao desenvolvedor focar seus esforços na resolução do problema em si, e não ficar reescrevendo software(código). Ou seja os framework ajudam o desenvolvedor a projetar o sistema de forma mais eficiente tem uma curva de aprendizagem relativamente fácil, e a reutilização de códigos que geram menos esforço dos programadores pelo fato de não ter que inventar a roda novamente vasta reutiliza, então ele tem uma vasta importância, Basicamente, é um template com diversas funções que podem ser usadas pelo desenvolvedor. Com ele, é desnecessário gastar tempo para reproduzir a mesma função em diferentes projetos.

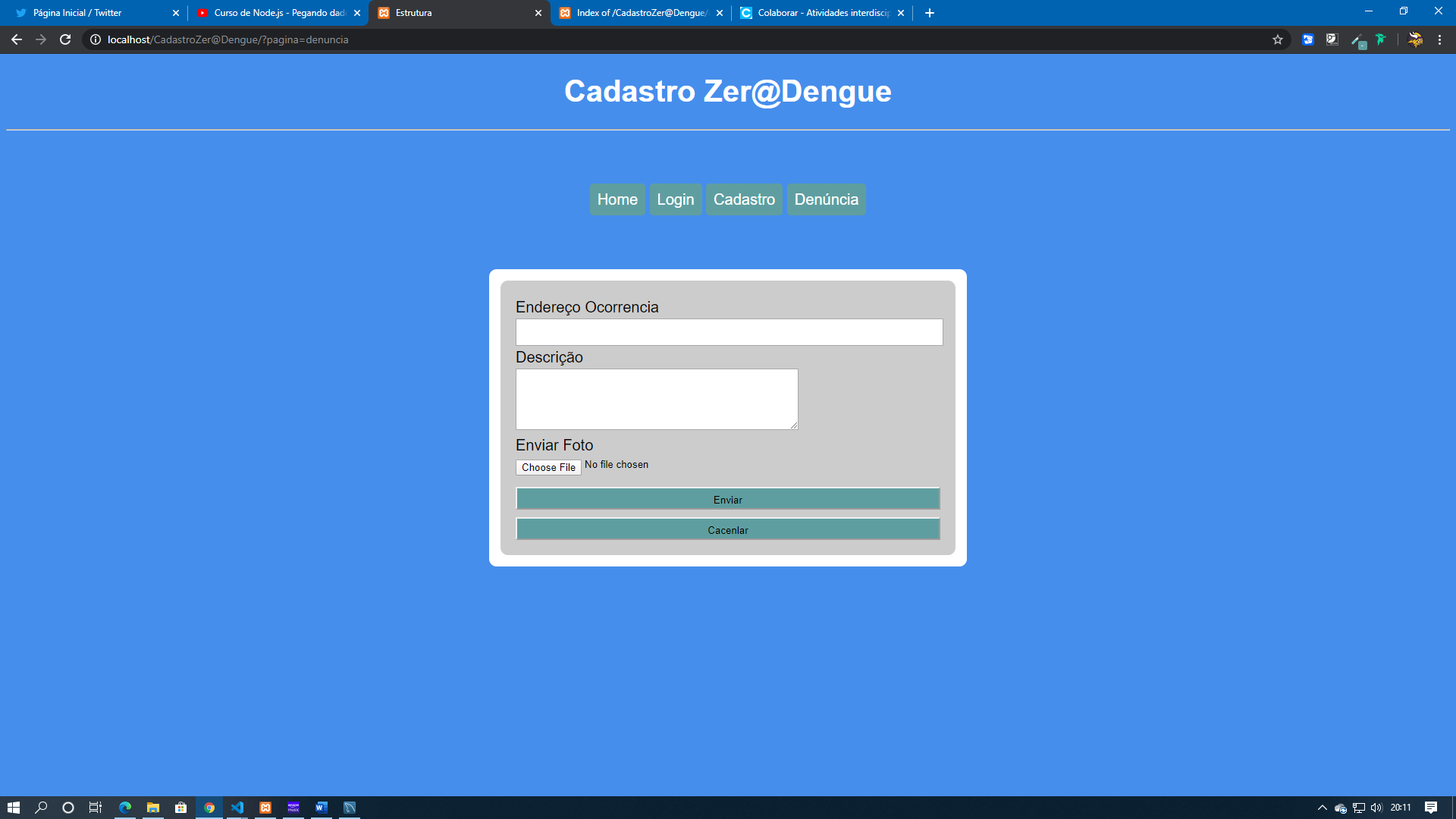
2.3- O Laravel oferece aos usuários 3 opções de painéis para administrar os projetos: Horizon, Nova e Echo. Oferece um kit de ferramentas muito amplo e opções que facilitam o trabalho em equipe, o que é particularmente desejável em projetos com mais de um programador. A parte visual e estética também tem importância, porém mais orientada ao quão fácil é a edição do conteúdo.

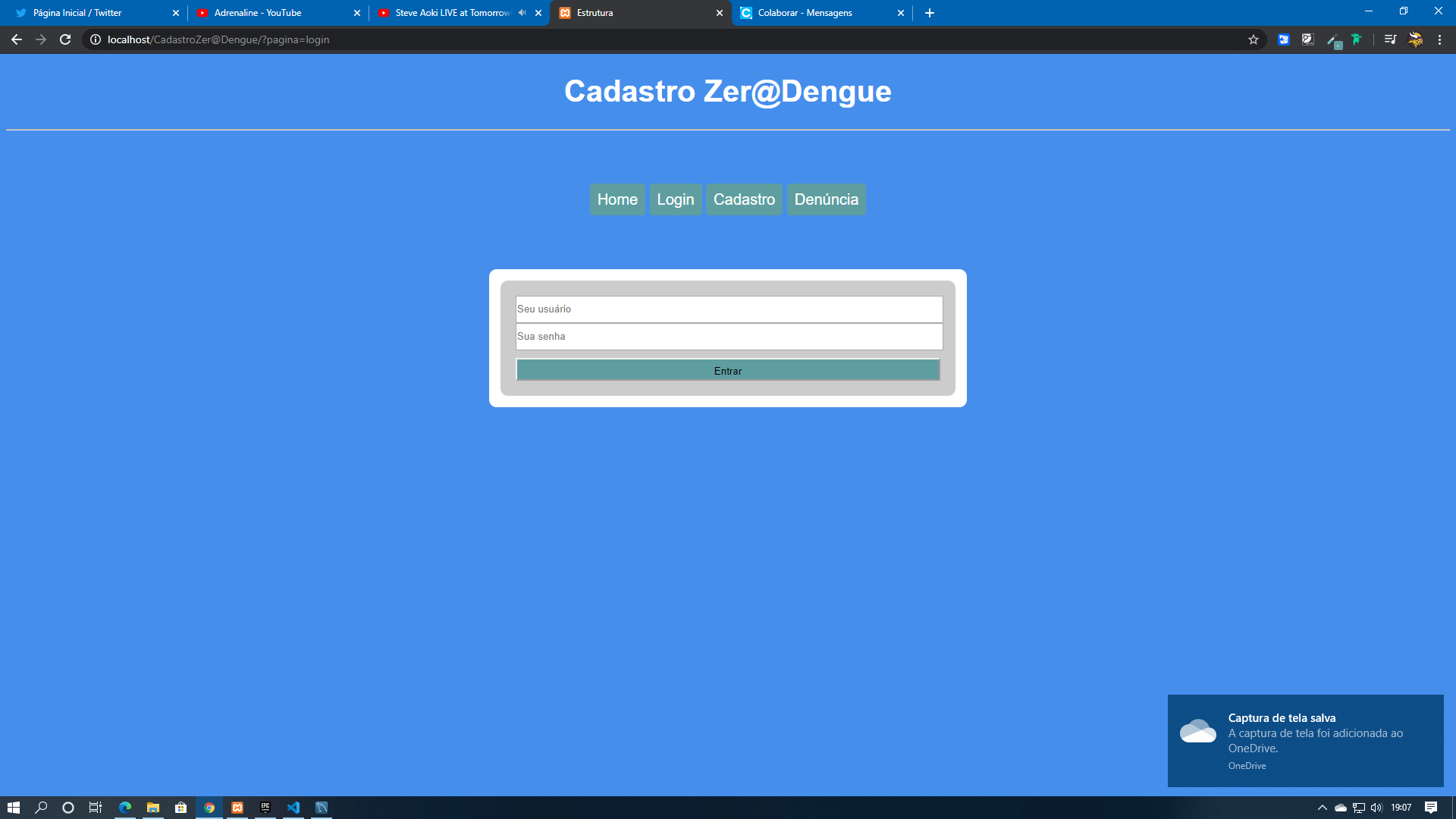
2.4- Zend Framework. O Zend é orientado a performance, sendo que a versão mais recente oferece um desempenho até quatro vezes mais rápido que a anterior. Utiliza uma arquitetura aberta que tanto pode ser baseada em Middleware ou MVC, seguindo os padrões PHP-FIG. Também privilegia aspectos de segurança, como o armazenamento de senhas usando bcrypt ou gera criptografia usando AES.

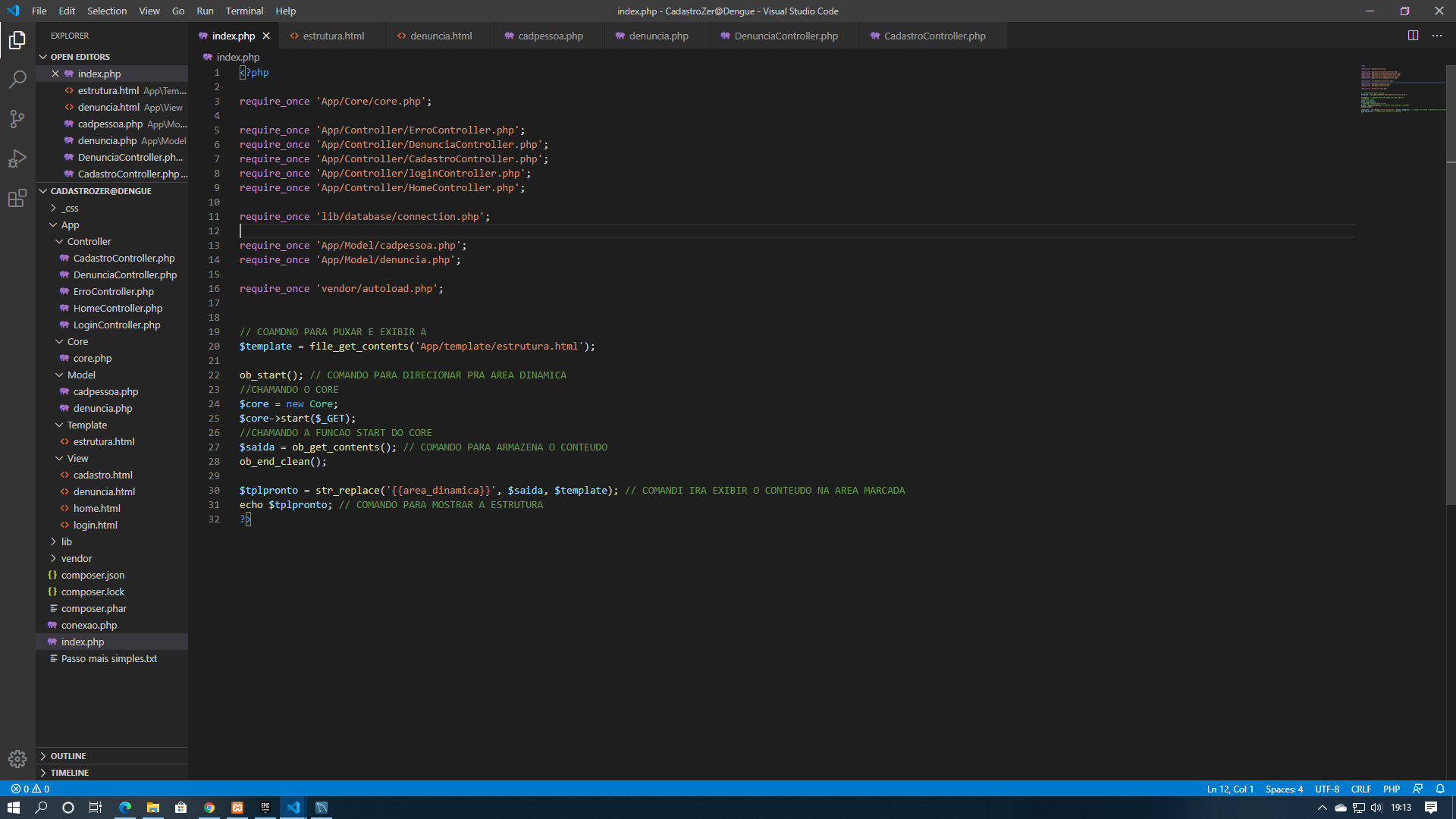
2.5- Um padrão que poderia se encaixar seria o padrão abstract factory ele e um padrão de projeto cri acional, que resolve o problema de criar famílias de produtos sem especificar suas classes concretas. O Abstract Factory define uma interface para criar todos os produtos distintos, mas deixa a criação real do produto para classes fábrica concretas. Cada tipo de fábrica corresponde a uma determinada variedade de produtos. Portanto como e um sistema de cadastro se encaixava bem com algumas alterações.

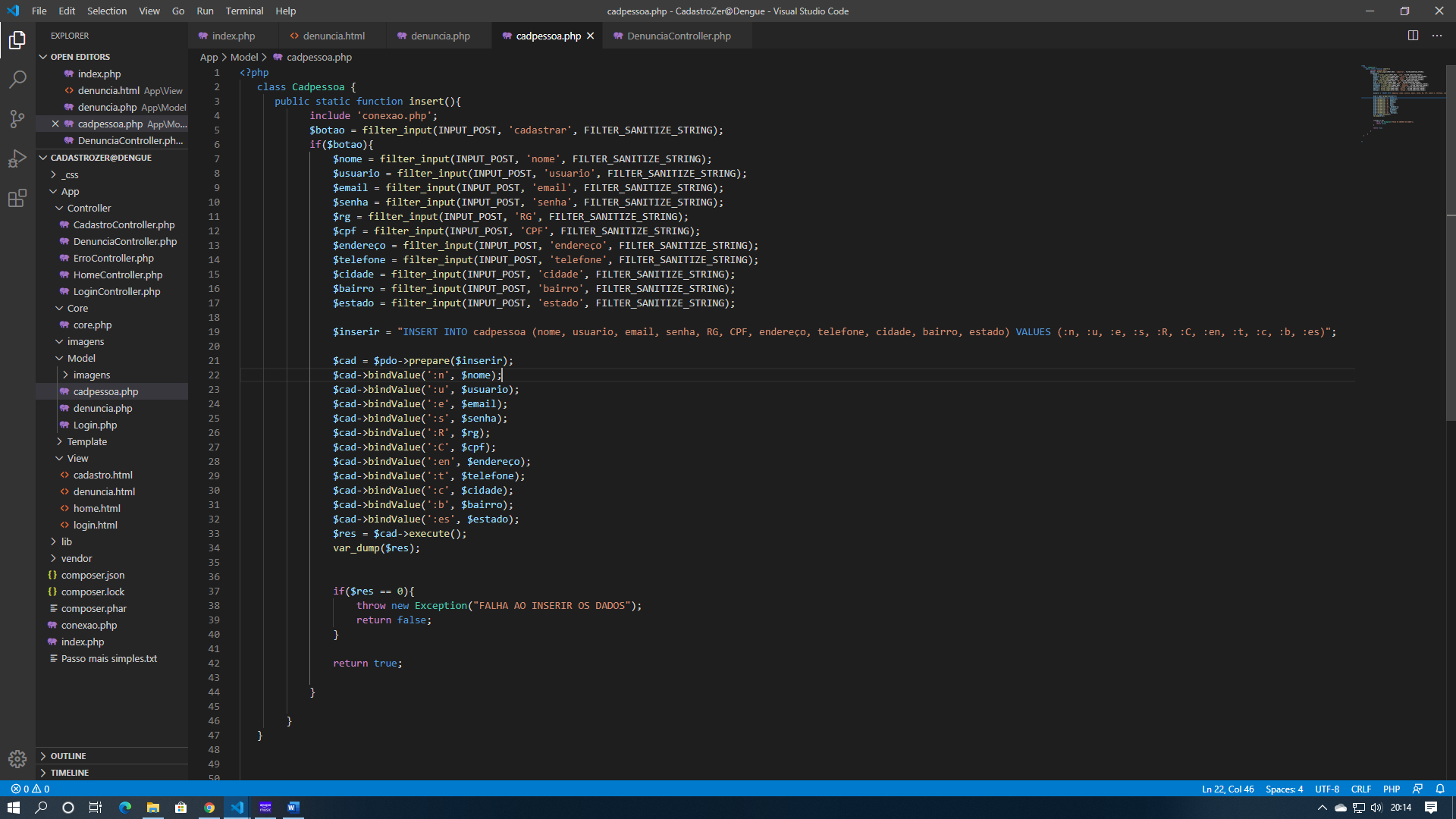
## **Tarefa 3: Programação para WEB II**

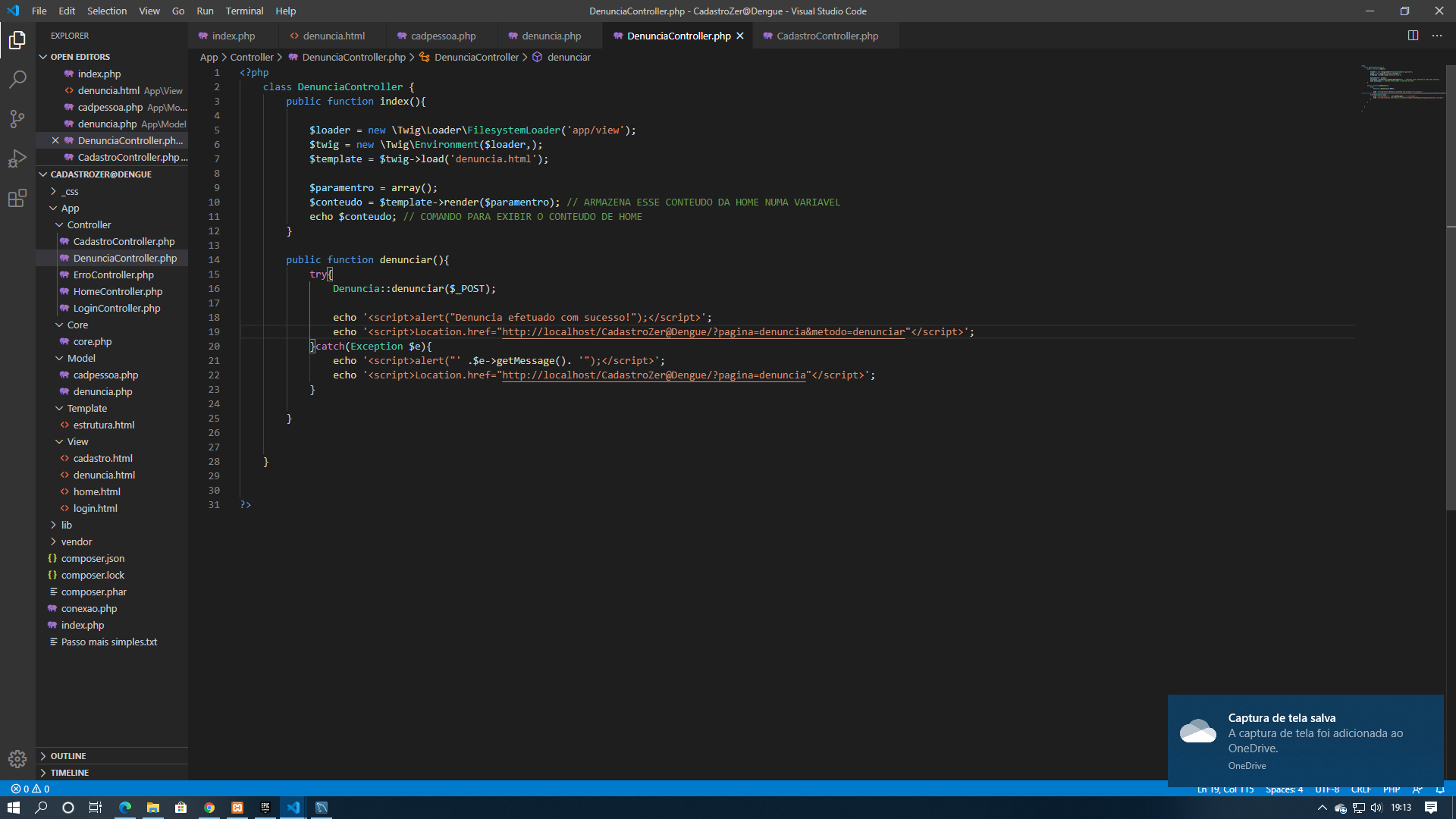
.

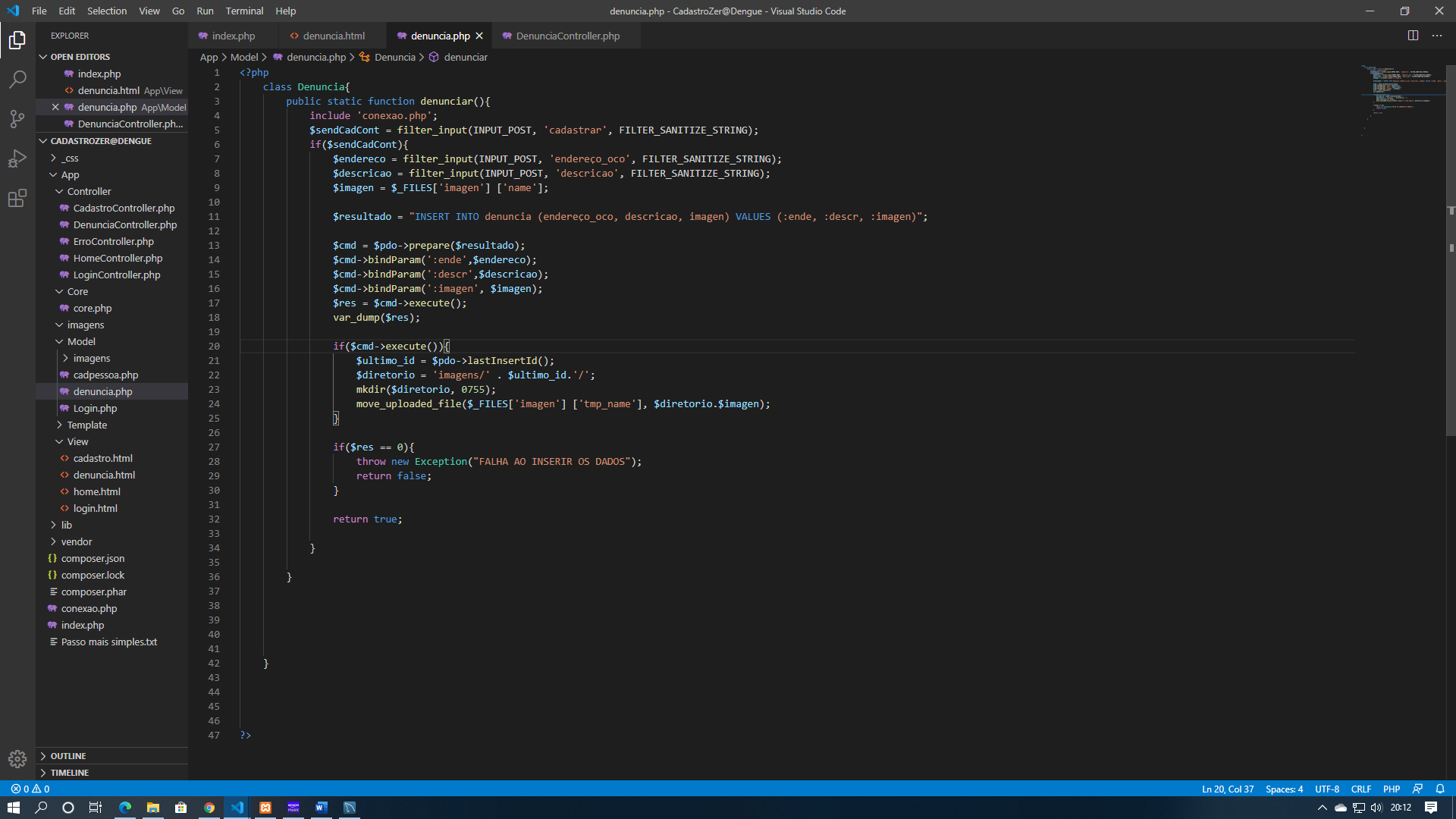


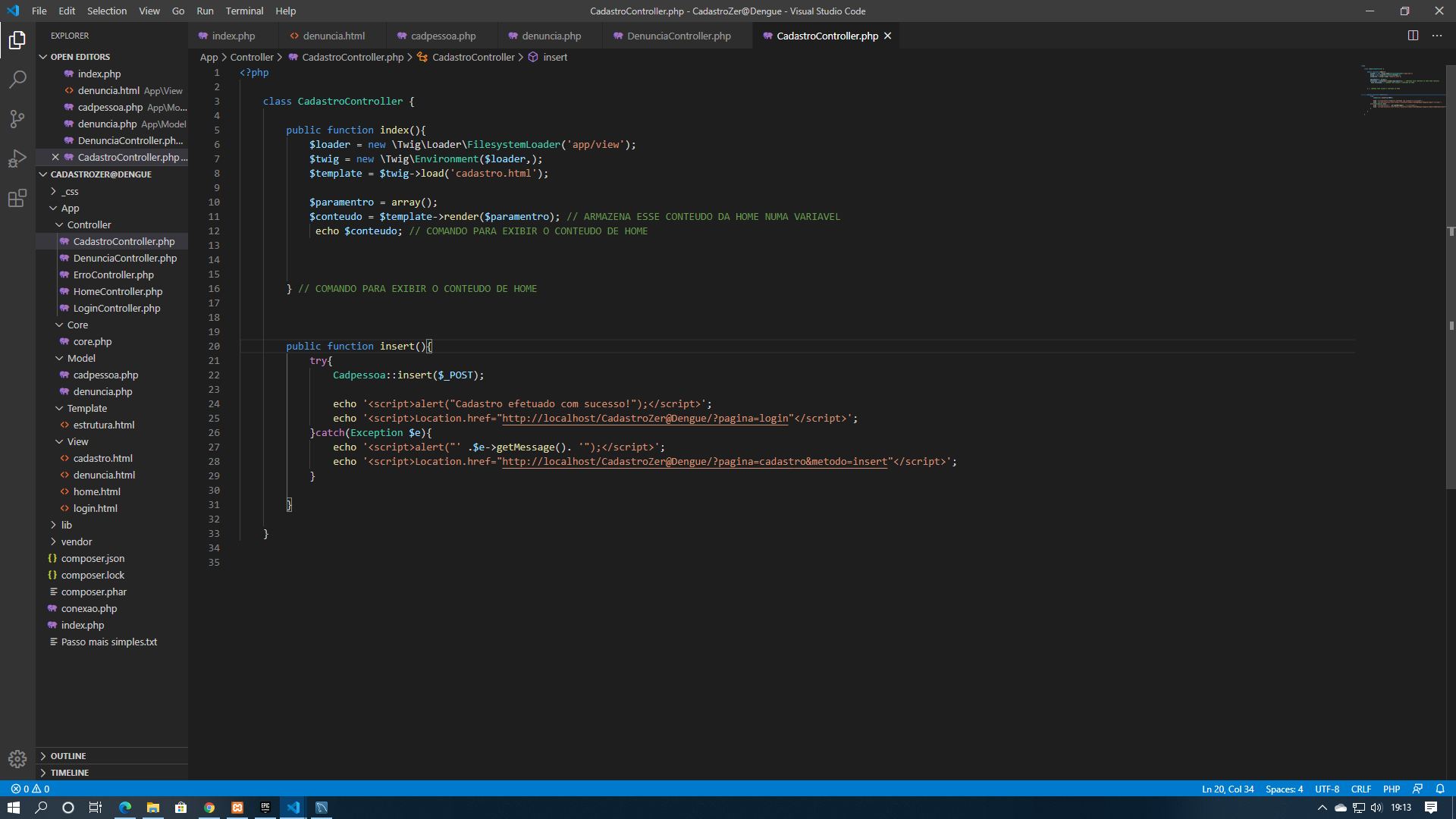






****

****



3.1**-** Documentação do sistema em funcionamento

Projeto foi desenvolvido usando o padrão MVC junto com a linguagem PHP + COMPOSER + TWIG/TWIG para trazer o conteúdo dinamicamente , foram criados as classe denuncia e a classe cadastro, também foi criado as controller para cada respectiva pagina, eu criei também os métodos de inserção, denunciar e verificalogin que são responsáveis em fazer a inserção no banco de dados caso requisitado, desenvolvi também a página de login e home. Foi criado um controller para página de erro caso a pessoa digite a pagina errada retornara um erro. Por tanto como funciona esse sistema a pessoa acessa o site ou aplicativo faz o cadastro com seus dados depois do cadastro ser aprovado ela e redirecionada a pagina de login onde ela irá se logar com seu usuário e senha escolhido, caso o cadastro não seja cadastrado com sucesso volta pra pagina de cadastro. Depois da pessoa entrar no sistema com seu usuário e senha ela terá acesso a página denúncia onde poderá efetuar sua denúncia que também retornara uma mensagem se foi efetuado com sucesso ou não.

**CONCLUSÃO**

Bom a conclusão e que esse tema proposto e interessante pelo fato de ser rotina na vida de um desenvolvedor/analista que estará desenvolvendo sistema como esse que nos foi dado, então procurei usar todo o conhecimento que tive nas matérias desse semestre mais as vídeo aulas do portal mais conteúdo de terceiros, também apliquei meu conhecimento, minha lógica no desenvolvimento e espero ter colocado tudo que foi pedido, todo o conteúdo desenvolvido estará disponível no github que deixarei o link nas referencias com o nome do repositório portfolio5.

**REFERÊNCIAS BIBLIOGRÁFICAS**

<https://astah.net/downloads/>

<https://www.devmedia.com.br/quick-tips-padroes-de-projeto-no-php/14452>

<https://micreiros.com/padroes-de-arquiteturas-mvc-mvp-e-pipeline/><https://sourceforge.net/projects/brmodelo30/>

<https://code.visualstudio.com/>

<https://www.apachefriends.org/pt_br/index.html>

<https://github.com/MaykonRegis/Portfolio5>

<https://getcomposer.org/>

<https://packagist.org/>

**Todos os livros do Semestre**