FraudWatch «model» **(C**) Usuario «controller» «record» «record» EnderecoResponse UsuarioRequest EnderecoRequest □id: Long (C) UsuarioController (C) EnderecoController «config» □ telefone: String o cep(): String o telefone(): String □ nome: String **(C**) Cidade o cidade(): String (C) Regiao **C**) AppConfig o cidade(): String o email(): String o CreateEnderecos(List<EnderecoRequest>): ResponseEntity<Map<String, Object>> o GetAllUsuarios(int): ResponseEntity<List<UsuarioResponse>> □ sobrenome: String (C) Inicializador Dados o numero(): String o id(): Long o tipoUsuarioid(): Long CreateEndereco(EnderecoRequest): ResponseEntity<EnderecoResponse> □ nomeCidade: String □ dataCadastro: LocalDateTime CreateUsuario(UsuarioReguest): ResponseEntity<UsuarioResponse> o complemento(): String □id: Long o ReadEnderecoById(Long): ResponseEntity<EnderecoResponse> o senha(): String o regiao(): String o restTemplate(): RestTemplate o CreateUsuarios(List<UsuarioRequest>): ResponseEntity<List<UsuarioResponse>> □id: Long □ email: String o cep(): String o init(): void □ nomeRegiao: String o DeleteEndereco(Long): ResponseEntity < EnderecoResponse > o GetUsuarioById(Long): ResponseEntity<UsuarioResponse> o complemento(): String o sobrenome(): String □ estado: Estado o executorService(): ExecutorService □ cpf: String o bairro(): String o dataNascimento(): LocalDate o UpdateUsuarioById(Long, UsuarioRequest): ResponseEntity<UsuarioResponse> o estado(): String o UpdateEnderecoById(Long, EnderecoRequest): ResponseEntity<EnderecoResponse> □ senha: String o regiao(): String o cpf(): String o numero(): String o ReadAllEnderecos(int): ResponseEntity<List<EnderecoResponse>> o DeleteUsuario(Long): ResponseEntity<UsuarioResponse> □ dataNascimento: LocalDate o estado(): String o nome(): String o logradouro(): String □ endereco: Endereco o logradouro(): String o bairro(): String o endereco(): EnderecoRequest □ tipoUsuario: TipoUsuario associate E UsuarioResponse «service» **(C**) Endereco o endereco(): EnderecoResponse √ **E** ViaCepResponse o telefone(): String © EnderecoService (C) UsuarioService □ cep: String **(C)** TipoUsuario **C**) EnderecoServiceAsync o nome(): String **(C)** TipoUsuarioService o localidade(): String □ complemento: String o sobrenome(): String o bairro(): String o requestToEndereco(EnderecoRequest, Future<ViaCepResponse>): Endereco □ bairro: Bairro o updateUsuario(Long, UsuarioRequest, EnderecoRequest): Usuario □ descricao: String o obterEnderecoPorCepAsync(String): Future < ViaCepResponse > o dataCadastro(): LocalDateTime o usuarioToResponse(Usuario): UsuarioResponse o cep(): String □ capitalizeWords(String): String □ logradouro: String o inicializarTiposUsuario(): void □id: Long o obterEnderecoPorCep(String): ViaCepResponse o dataNascimento(): LocalDate o estado(): String o enderecoToResponse(Endereco): EnderecoResponse requestToUsuario(UsuarioRequest, EnderecoRequest): Usuario □id: Long o cpf(): String o regiao(): String □ numero: String use use o email(): String o logradouro(): String oid(): Long o senha(): String o tipoUsuario(): String associate *«exception»* «repository (C) ValidationExceptionHandler **(C**) Bairro **(C**) Estado «interface»
CidadeRepository UsuarioRepository *«interface»* (I) RegiaoRepository «interface»
EnderecoRepository EstadoRepository BairroRepository (C) CepNaoEncontradoException (C) CpfJaCadastradoException (C) EmailJaCadastradoException o handleCpfJaCadastradoException(CpfJaCadastradoException): ResponseEntity<String> □ nomeBairro: String □id: Long o handleCepNaoEncontradoException(CepNaoEncontradoException): ResponseEntity<String> □id: Long □ nomeEstado: String o findByCpf(String): Optional<Usuario> o findByNomeRegiao(String): Optional<Regiao> o findByNomeEstado(String): Optional<Estado> o findByNomeBairro(String): Optional<Bairro> o findByNomeCidade(String): Optional<Cidade> □ cidade: Cidade □regiao: Regiao o handleValidationExceptions(MethodArgumentNotValidException): ResponseEntity<Map<String, String>> ofindByEmail(String): Optional<Usuario> o handleEmailJaCadastradoException(EmailJaCadastradoException): ResponseEntity < String > TipoUsuarioRepository o findByDescricao(String): Optional<TipoUsuario>