



Arquitetura de Software

--- Grupo 4 ---

Winicius Silva

Jorge Uliam

Pedro Duo

Samuel Lopes

Kaue Volpe

Carlos Mascena



Desenho da Solução - Containers

VAMOS DAR ZOOM NO WEB APPLICATION



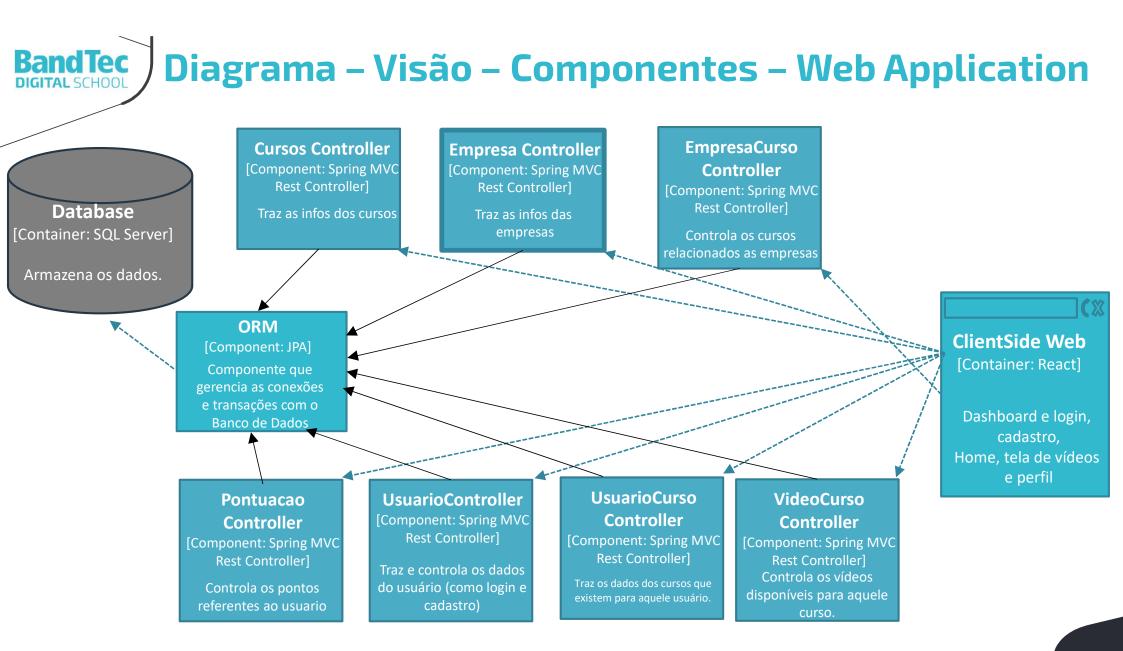




Diagrama de Classes - Curso Controller

<<Java Class>>
CursoController

+ getCurso():

ResponseEntity

+ getCursoCategoria():

ResponseEntity

+getCursosSearch(String):

ResponseEntity

+getCursosSubCategoria(String):

ResponseEntity

+getOneCurso(Integer):

ResponseEntity

<<Java Class>>
CursoRepository

+ findByCategoria(String):

List<Curso>

+findBySubCategoria(String):

List<Curso>

<<Java Class>>

InternalResourceViewResolver

+ viewResolver()

<<Java Class>
Curso (Model)

idCurso: IntegernomeCurso: String

- categoria: String

- subCategoria: String

- preco: Double

- descricao : String

- qtdPontos : Integer

+ getIdCurso(): Integer

+ getNomeCurso(): String

+ getCategoria(): String

+ getSubCategoria(): String

+ getPreco(): Double

+ getDescricao(): String

+ getQtdPontos():Integer



Diagrama de Classes - Empresa Controller

<<Java Class>> **EmpresaRepository** <<Java Class>> **EmpresaController** <<Java Class>> **InternalResourceViewResolver** + viewResolver()

<<Java Class>
Empresa (Model)

- idEmpresa: Integer

- nome: String

- cnpj: String

- endereco: String

- email: String

- senha: String

- duracaoDoContrato : String

- valorDoContrato : Double

+ getIdEmpresa(): Integer

+ getNome(): String

+ getCnpj(): String

+ getEndereco(): String

+ getEmail(): String

+ getSenha(): String

+ getDuracaoDoContrato():

String

+ getValorDoContrato(): Doube



Diagrama de Classes – EmpresaCurso Controller

<<Java Class>>

 ${\bf Empresa Curso Controller}$

+postEmpresaCurso (EmpresaCurso):

ResponseEntity

+ getEmpresaByCurso(Integer):

Response Entity

<<Java Class>>

EmpresaCursoRepository

<<Java Class>>

InternalResourceViewResolver

+ viewResolver()

<<Java Class>
EmpresaCurso (Model)

- idEmpresaCurso: Integer

- fkEmpresa: Empresa

- fkCurso: Curso

- data: LocalDate

+ getIdEmpresaCurso(): Integer

+ getFkEmpresa(): Empresa

+ getFkCurso(): Curso

+ getData(): LocalDate



Diagrama de Classes - Pontuacao Controller

<<Java Class>>

PontuacaoController

- +getPontos (): ResponseEntity
- + postPontos (Pontuacao):

ResponseEntity

+getPontosPorUsuario (Integer):

ResponseEntity

+ getPontosTotalPorUsuario (Integer):

ResponseEntity

<<Java Class>>

PontuacaoRepository

+ findAllByFkUsuario(Integer): List<Pontuacao>

<<Java Class>>

InternalResourceViewResolver

+ viewResolver()

<<Java Class> Pontuacao (Model)

- idPontosUsuario: Integer
- data: LocalDate
- pontos: Double
- fkUsuario: Usuario
- fkCurso: Curso
- + getIdPontosUsuario(): Integer
- + getData(): LocalDate
- + getPontos(): Double
- + getFkUsuario(): Usuario
- + getFkCurso(): Curso



Diagrama de Classes - Usuario Controller

<<Java Class>> UsuarioController

+getUsuario (): ResponseEntity

+postCadastrar (Usuario):

ResponseEntity

+getUsuario (Integer): ResponseEntity

+getLogin (String,

String):ResponseEntity

<<Java Class>> UsuarioRepository

Usuario

+ findByEmailAndSenha(String, String):

<<Java Class>>

InternalResourceViewResolver

+ viewResolver()

<<Java Class> Usuario (Model)

- idUsuario: Integer

- nomeUsuario: String

- cpf: String

- dataNascimento: LocalDate

- celular: String

- email: String

- horaCadastro : LocalDate

- senha: String

- FkEmpresa: Empresa

+ getIdUsuario(): Integer

+ getNomeUsuario(): String

+ getCpf(): String

+ getDataNascimento():

LocalDate

+ getCelular(): String

+ getEmail(): String

+ getHoraCadastro: LocalDate

+ getSenha(): String

+ getFkEmpresa(): Empresa



ponseEntity

Diagrama de Classes – UsuarioCurso Controller

<<Java Class>> UsuarioCursoController

+getMyCurso(Integer):ResponseEntity
+getCursoFinalizados(Integer):
ResponseEntity
+patchCursosFinalizados(Double,
Integer, Integer):ResponseEntity
+postRecentCurso(UsuarioCurso):Res

<<Java Class>> UsuarioCursoRepository

+ atualizarFinalizado(Integer, Integer, Interger):UsuarioCurso +atualizarProgresso(Double, Integer, Integer):UsuarioCurso +atualizarDate(Date, Integer, Integer):UsuarioCurso

<<Java Class> Usuario (Model)

- idUsuarioCurso: Integer
- fkUsuario: Usuario
- fkCurso: Curso
- finalizado: Integer
- date: LocalDate
- progresso : Double
- + getIdUsuarioCurso (): Integer
- + getFkUsuario (): Usuario
- + getFkCurso (): Curso
- + getFinalizado (): Integer
- + getDate (): LocalDate
- + getProgresso (): Double

<<Java Class>>

InternalResourceViewResolver

+ viewResolver()



Diagrama de Classes - VideoCurso Controller

<<Java Class>>

VideoCursoController

+getVideoByCurso (Integer):

ResponseEntity

+postVideoByCurso (VideoCurso):

ResponseEntity

<<Java Class>>

VideoCursoRepository

<<Java Class>>

InternalResourceViewResolver

+ viewResolver()

<<Java Class>

VideoCurso (Model)

- idVideoCurso: Integer

- fkCurso: Curso

- Titulo: String

- Indice:Int

- link: String

- descricaoVideo: String

+ getIdVideoCurso (): Integer

+ getFkCurso (): Curso

+ getLink (): String

+ getDescricaoVideo (): String