

SCD - Sistema de Capacitação à Distância
Especificação de Caso de Uso
Acesso aos conteúdos

Versão 1.0
Autor(es):
Bruno Perotti
Camila Soika
Elias Augusto

Índice Analítico

1. Controle
 - 1.1 Histórico da Revisão
 - 1.2 Referências
 - 1.3 Definições, acrônimos e abreviações
2. Introdução
 - 2.1 Finalidade
 - 2.2 Escopo
3. Representação da arquitetura
4. Metas e restrições de arquitetura
5. Visão de casos de uso
 - 5.1 Realizações de casos de uso
6. Visão lógica
 - 6.1 Visão geral
 - 6.2 Pacotes de design significativos do ponto de vista da arquitetura
7. Visão de processos
8. Visão de implantação
9. Visão de implementação
 - 9.1 Visão geral
 - 9.2 Camadas
10. Visão de dados
11. Tamanho e desempenho
12. Qualidade

Documento de Arquitetura de Software

1. Controle

1.1 Histórico da Revisão

Data	Versão	Descrição	Autor
14/10/2014	1.0	Modelo lógico	Camila
14/10/2014	1.0	Dicionário	Bruno
16/10/2014	1.0	Modelo conceitual e físico	Camila

1.2 Referências

Título	Versão	Status	Autor
Visão	1.8	Andamento	Bruno, Camila e Elias
Diagrama de classe	1.3	Andamento	Bruno

1.3 Definições, acrônimos e abreviações

Verificar Glossário

2.Introdução

2.1 Finalidade

Este documento apresenta uma visão geral abrangente da arquitetura do sistema e utiliza uma série de visões arquiteturais diferentes para ilustrar os diversos aspectos do sistema. Sua intenção é capturar e transmitir as decisões significativas do ponto de vista da arquitetura que foram tomadas em relação ao sistema.

2.2 Escopo

3.Representação da Arquitetura

Este documento apresenta a arquitetura como uma série de visões: visão de casos de uso, visão de processos, visão de implantação e visão de implementação.

4.Metas e Restrições de Arquitetura

5.Visão de Casos de Uso

5.1 Realizações de Casos de Uso

6.Visão Lógica

6.1 Visão Geral

6.2 Pacotes de Design Significativos do Ponto de Vista da Arquitetura

7.Visão de Processos

8.Visão de Implantação

9.Visão de Implementação

9.1 Visão Geral

9.2 Camadas

10.Visão de Dados

10.1 Descrição das Entidades

Nome Entidade: Usuário			
Descrição: Utilizadores da aplicação			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do usuário	Não nulo
Nome	Texto	Nome completo de uma pessoa	Não nulo
Email	Texto	Email completo do usuário	Não nulo
login	Texto	Acesso a aplicação	Não nulo

Nome Entidade: Setor			
Descrição: Lotação do usuário			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador o setor	Não nulo
Nome	Texto	Nome completo do setor do usuário	Não nulo

Nome Entidade: Tipo de usuário			
Descrição: nível de acesso na aplicação			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do tipo de usuário	Não nulo

Nome	Texto	Nome completo do setor do usuário	Não nulo
------	-------	-----------------------------------	----------

Nome Entidade: Curso			
Descrição: Curso disponibilizado na aplicação			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do curso	Não nulo
Nome	Texto	Nome completo do curso	Não nulo
Imagem	Texto	Imagem demonstrativa do curso	Nulo
Descrição	Texto	Descrição do curso	Não nulo

Nome Entidade: Status_curso			
Descrição: Status do curso - desempenho			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do status do curso	Não nulo
Status	Texto	Status do curso	Não nulo

Nome Entidade: modulo			
Descrição: módulo que está vinculado ao curso			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do módulo	Não nulo
Nome	Texto	Nome do módulo	Não nulo
Desempenho	Texto	Desempenho do módulo, caso seja avaliativo	Nulo

Nome Entidade: avaliacao			
Descrição: Avaliação – prova – caso o módulo seja avaliativo			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador	Não nulo

		da avaliação	
Descrição	Texto	Descrição da avaliação	Não nulo
Nota	Texto	Nota tirada pelo usuário	Nulo

Nome Entidade: questao			
Descrição: pergunta vinculada a avaliação			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador da questão	Não nulo
Descrição	Texto	Descrição da questão - pergunta	Não nulo
Nota	Texto	Nota tirada pelo usuário	Nulo

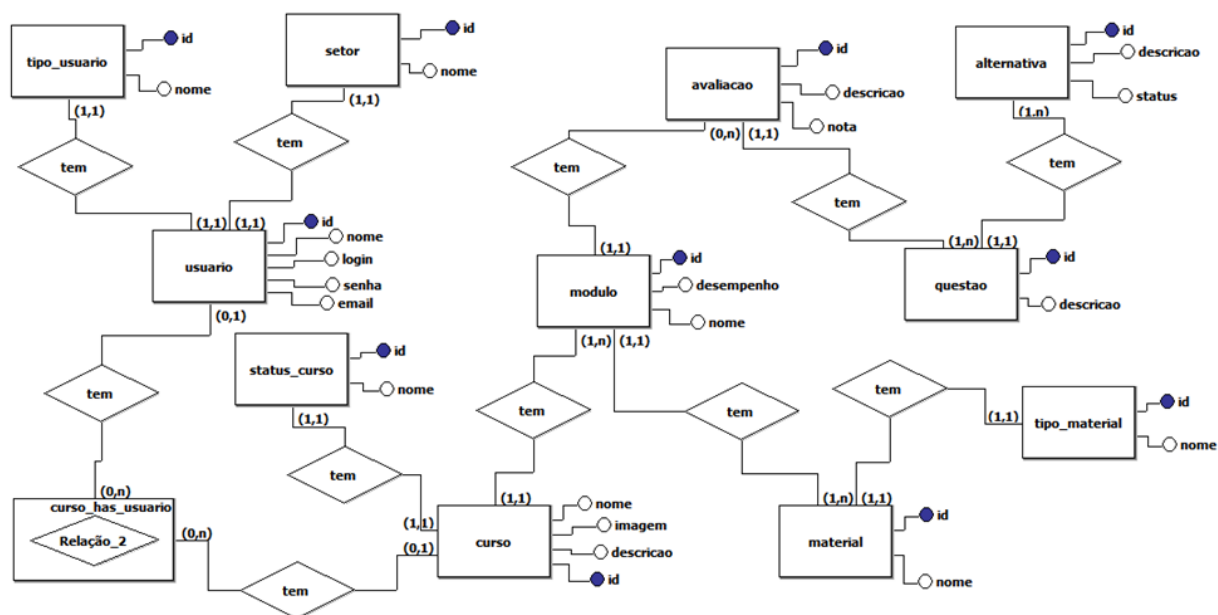
Nome Entidade: alternativa			
Descrição: alternativas vinculadas a questão			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador da alternativa	Não nulo
Descrição	Texto	Descrição da alternativa	Não nulo
Status	Texto	Alternativa correta	Não Nulo

Nome Entidade: material			
Descrição: conteúdo inserido no módulo desejado			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do conteúdo inserido	Não nulo
nome	Texto	Descrição da alternativa	Não nulo

Nome Entidade: tipo_material			
Descrição: tipo de material inserido (video, texto, imagem, etc...).			
Atributos	Tipo do Campo	Descrição	Observações
Id	Numérico	Valor único identificador do conteúdo inserido	Não nulo
nome	Texto	Material inserido no banco	Não nulo

10.2 Diagrama Entidade Relacionamento

*Utilizar o software BrModelo



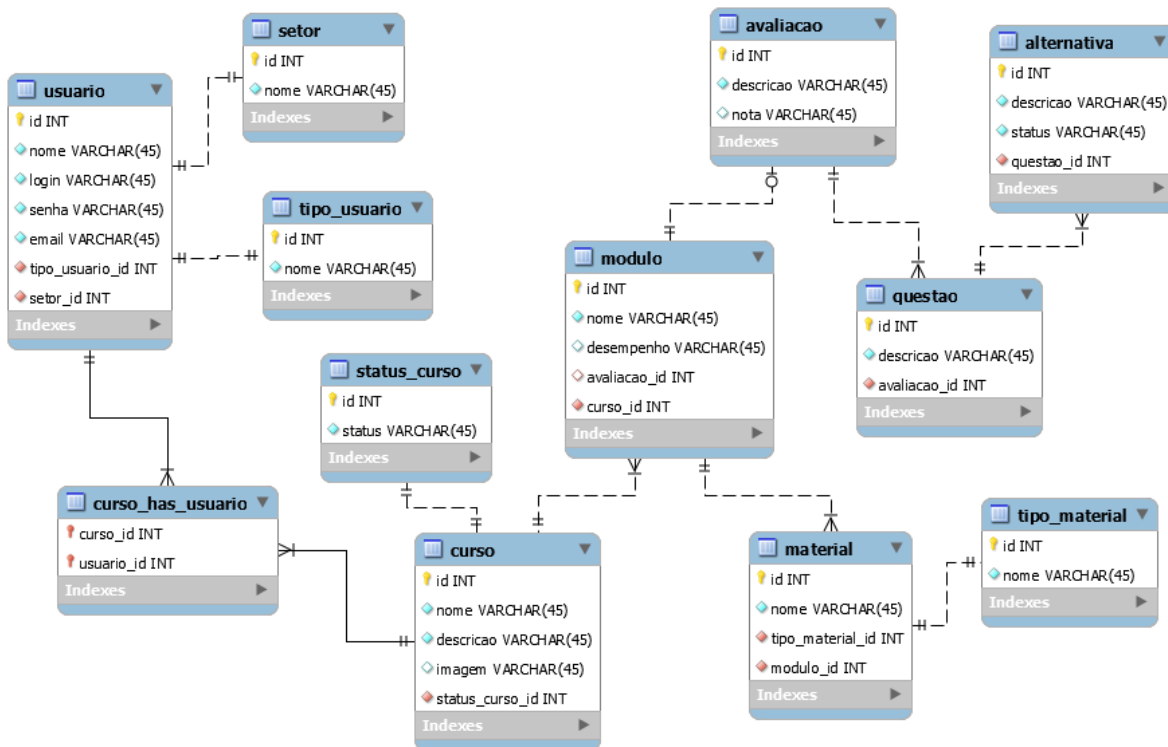
Versão 1.0

Data: 16/10/2014

10.3 Modelo lógico

*Utilizar o software Workbench

Versão 1.0



Data: 14/10/2014

Responsável: Bruno e Camila

10.4 Modelo Físico

-- Schema ProjetoSCD

```
CREATE SCHEMA IF NOT EXISTS `ProjetoSCD` DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci ;
```

-- Table `ProjetoSCD`.`tipo_usuario`

```
DROP TABLE IF EXISTS `ProjetoSCD`.`tipo_usuario` ;
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`tipo_usuario` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`id`))
```

-- Table `ProjetoSCD`.`setor`

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`setor` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`id`))
```

```
`id` INT NOT NULL AUTO_INCREMENT,  
`nome` VARCHAR(45) NOT NULL,  
PRIMARY KEY (`id`))
```

```
-- Table `ProjetoSCD`.`usuario`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`usuario` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NOT NULL,  
  `login` VARCHAR(45) NOT NULL,  
  `senha` VARCHAR(45) NOT NULL,  
  `email` VARCHAR(45) NOT NULL,  
  `tipo_usuario_id` INT NOT NULL,  
  `setor_id` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `fk_usuario_tipo_usuario_idx` (`tipo_usuario_id` ASC),  
  INDEX `fk_usuario_setor1_idx` (`setor_id` ASC),  
  CONSTRAINT `fk_usuario_tipo_usuario`  
    FOREIGN KEY (`tipo_usuario_id`)  
      REFERENCES `ProjetoSCD`.`tipo_usuario` (`id`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION,  
  CONSTRAINT `fk_usuario_setor1`  
    FOREIGN KEY (`setor_id`)  
      REFERENCES `ProjetoSCD`.`setor` (`id`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION)
```

```
-- Table `ProjetoSCD`.`status_curso`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`status_curso` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `status` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`id`))
```

```
-- Table `ProjetoSCD`.`curso`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`curso` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `nome` VARCHAR(45) NOT NULL,  
  `descricao` VARCHAR(45) NOT NULL,  
  `imagem` VARCHAR(45) NULL,  
  `status_curso_id` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `fk_curso_status_curso1_idx` (`status_curso_id` ASC),  
  CONSTRAINT `fk_curso_status_curso1`  
    FOREIGN KEY (`status_curso_id`)  
      REFERENCES `ProjetoSCD`.`status_curso` (`id`)  
      ON DELETE NO ACTION  
      ON UPDATE NO ACTION)
```

```
-- Table `ProjetoSCD`.`avaliacao`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`avaliacao` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `descricao` VARCHAR(45) NOT NULL,
  `nota` VARCHAR(45) NULL,
  PRIMARY KEY (`id`))
```

```
-- Table `ProjetoSCD`.`modulo`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`modulo` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `nome` VARCHAR(45) NOT NULL,
  `desempenho` VARCHAR(45) NULL,
  `avaliacao_id` INT NULL,
  `curso_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_modulo_avaliacao1_idx` (`avaliacao_id` ASC),
  INDEX `fk_modulo_curso1_idx` (`curso_id` ASC),
  CONSTRAINT `fk_modulo_avaliacao1`
    FOREIGN KEY (`avaliacao_id`)
      REFERENCES `ProjetoSCD`.`avaliacao` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_modulo_curso1`
    FOREIGN KEY (`curso_id`)
      REFERENCES `ProjetoSCD`.`curso` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

```
-- Table `ProjetoSCD`.`tipo_material`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`tipo_material` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `nome` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`id`))
```

```
-- Table `ProjetoSCD`.`material`
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`material` (
  `id` INT NOT NULL AUTO_INCREMENT,
  `nome` VARCHAR(45) NOT NULL,
  `tipo_material_id` INT NOT NULL,
  `modulo_id` INT NOT NULL,
  PRIMARY KEY (`id`),
  INDEX `fk_material_tipo_material1_idx` (`tipo_material_id` ASC),
  INDEX `fk_material_modulo1_idx` (`modulo_id` ASC),
  CONSTRAINT `fk_material_tipo_material1`
    FOREIGN KEY (`tipo_material_id`)
      REFERENCES `ProjetoSCD`.`tipo_material` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_material_modulo1`
    FOREIGN KEY (`modulo_id`)
      REFERENCES `ProjetoSCD`.`modulo` (`id`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

```
-- -----  
-- Table `ProjetoSCD`.`questao`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`questao` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `descricao` VARCHAR(45) NOT NULL,  
  `avaliacao_id` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `fk_pergunta_avaliacao1_idx` (`avaliacao_id` ASC),  
  CONSTRAINT `fk_pergunta_avaliacao1`  
    FOREIGN KEY (`avaliacao_id`)  
    REFERENCES `ProjetoSCD`.`avaliacao` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)
```

```
-- -----  
-- Table `ProjetoSCD`.`alternativa`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`alternativa` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `descricao` VARCHAR(45) NOT NULL,  
  `status` VARCHAR(45) NOT NULL,  
  `questao_id` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  INDEX `fk_alternativa_pergunta1_idx` (`questao_id` ASC),  
  CONSTRAINT `fk_alternativa_pergunta1`  
    FOREIGN KEY (`questao_id`)  
    REFERENCES `ProjetoSCD`.`questao` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)
```

```
-- -----  
-- Table `ProjetoSCD`.`curso_has_usuario`  
-- -----
```

```
CREATE TABLE IF NOT EXISTS `ProjetoSCD`.`curso_has_usuario` (  
  `curso_id` INT NOT NULL,  
  `usuario_id` INT NOT NULL,  
  PRIMARY KEY (`curso_id`, `usuario_id`),  
  INDEX `fk_curso_has_usuario_usuario1_idx` (`usuario_id` ASC),  
  INDEX `fk_curso_has_usuario_curso1_idx` (`curso_id` ASC),  
  CONSTRAINT `fk_curso_has_usuario_curso1`  
    FOREIGN KEY (`curso_id`)  
    REFERENCES `ProjetoSCD`.`curso` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION,  
  CONSTRAINT `fk_curso_has_usuario_usuario1`  
    FOREIGN KEY (`usuario_id`)  
    REFERENCES `ProjetoSCD`.`usuario` (`id`)  
    ON DELETE NO ACTION  
    ON UPDATE NO ACTION)
```

```
-- -----  
-- Data for table `ProjetoSCD`.`tipo_usuario`  
-- -----
```

```
INSERT INTO `ProjetoSCD`.`tipo_usuario` (`id`, `nome`) VALUES (1, 'Administrador');  
INSERT INTO `ProjetoSCD`.`tipo_usuario` (`id`, `nome`) VALUES (2, 'Gestor');
```

```
INSERT INTO `ProjetoSCD`.`tipo_usuario` (`id`, `nome`) VALUES (3, 'GerCurso');
INSERT INTO `ProjetoSCD`.`tipo_usuario` (`id`, `nome`) VALUES (4, 'Aluno');
```

```
-- Data for table `ProjetoSCD`.`setor`
```

```
INSERT INTO `ProjetoSCD`.`setor` (`id`, `nome`) VALUES (1, 'TI');
```

```
-- Data for table `ProjetoSCD`.`usuario`
```

```
INSERT INTO `ProjetoSCD`.`usuario` (`id`, `nome`, `login`, `senha`, `email`, `tipo_usuario_id`, `setor_id`)
VALUES (2, 'Lidiane', 'lidiane@empresa.com.br', '1234', 'lidiane@empresa.com.br', 1, 1);
INSERT INTO `ProjetoSCD`.`usuario` (`id`, `nome`, `login`, `senha`, `email`, `tipo_usuario_id`, `setor_id`)
VALUES (1, 'Administrador', 'admin', 'admin', 'admin@empresa.com.br', 1, 1);
```

```
-- Data for table `ProjetoSCD`.`status_curso`
```

```
INSERT INTO `ProjetoSCD`.`status_curso` (`id`, `status`) VALUES (1, 'Concluido');
INSERT INTO `ProjetoSCD`.`status_curso` (`id`, `status`) VALUES (2, 'Andamento');
INSERT INTO `ProjetoSCD`.`status_curso` (`id`, `status`) VALUES (3, 'Aberto');
```

```
-- Data for table `ProjetoSCD`.`curso`
```

```
INSERT INTO `ProjetoSCD`.`curso` (`id`, `nome`, `descricao`, `imagem`, `status_curso_id`) VALUES (1, 'Curso
de Logica Nivel Avancado', 'Esse curso visa desenvolver a logica do aluno.', NULL, 2);
```

```
-- Data for table `ProjetoSCD`.`avaliacao`
```

```
INSERT INTO `ProjetoSCD`.`avaliacao` (`id`, `descricao`, `nota`) VALUES (1, 'Avaliacao Logica', NULL);
```

```
-- Data for table `ProjetoSCD`.`modulo`
```

```
INSERT INTO `ProjetoSCD`.`modulo` (`id`, `nome`, `desempenho`, `avaliacao_id`, `curso_id`) VALUES (1,
'Pegadinhas', NULL, 1, 1);
```

```
-- Data for table `ProjetoSCD`.`tipo_material`
```

```
INSERT INTO `ProjetoSCD`.`tipo_material` (`id`, `nome`) VALUES (1, 'Video');
INSERT INTO `ProjetoSCD`.`tipo_material` (`id`, `nome`) VALUES (2, 'PDF');
```

```
-- Data for table `ProjetoSCD`.`questao`
```

```
INSERT INTO `ProjetoSCD`.`questao` (`id`, `descricao`, `avaliacao_id`) VALUES (1, 'Qual a cor do cavalo
branco de Napoleao?', 1);
```

```
-- -----  
-- Data for table `ProjetoSCD`.`alternativa`  
-- -----
```

```
INSERT INTO `ProjetoSCD`.`alternativa` (`id`, `descricao`, `status`, `questao_id`) VALUES (1, 'Roxo',  
'Falso', 1);  
INSERT INTO `ProjetoSCD`.`alternativa` (`id`, `descricao`, `status`, `questao_id`) VALUES (2, 'Azul',  
'Falso', 1);  
INSERT INTO `ProjetoSCD`.`alternativa` (`id`, `descricao`, `status`, `questao_id`) VALUES (3, 'Rosa',  
'Falso', 1);  
INSERT INTO `ProjetoSCD`.`alternativa` (`id`, `descricao`, `status`, `questao_id`) VALUES (4, 'Branco',  
'Verdadeiro', 1);
```

Versão 1.0

Data: 30/04/2014

Responsável: Lidiane Visintin

11.Tamanho e Desempenho

12.Qualidade