

Sistema de Distribuição de Disciplinas - UFG
Projeto de Arquitetura
versão 1.0.0

SDD-UFG	Versão: 1.0.0
Projeto de arquitetura	Data: 30/11/2015

Histórico da revisão

Data	Versão	Descrição	Autor(es)
30/11/2015	1.0.0	<p>→ Criada visão geral e detalhada de cada componente da arquitetura</p> <p>→ Adicionados diagramas da arquitetura e de classes com suas devidas explicações.</p>	Bruno Soares da Silva

SDD-UFG	Versão: 1.0.0
Projeto de arquitetura	Data: 30/11/2015

1. Visão geral

De modo geral, o sistema possui uma arquitetura composta de 3 componentes:

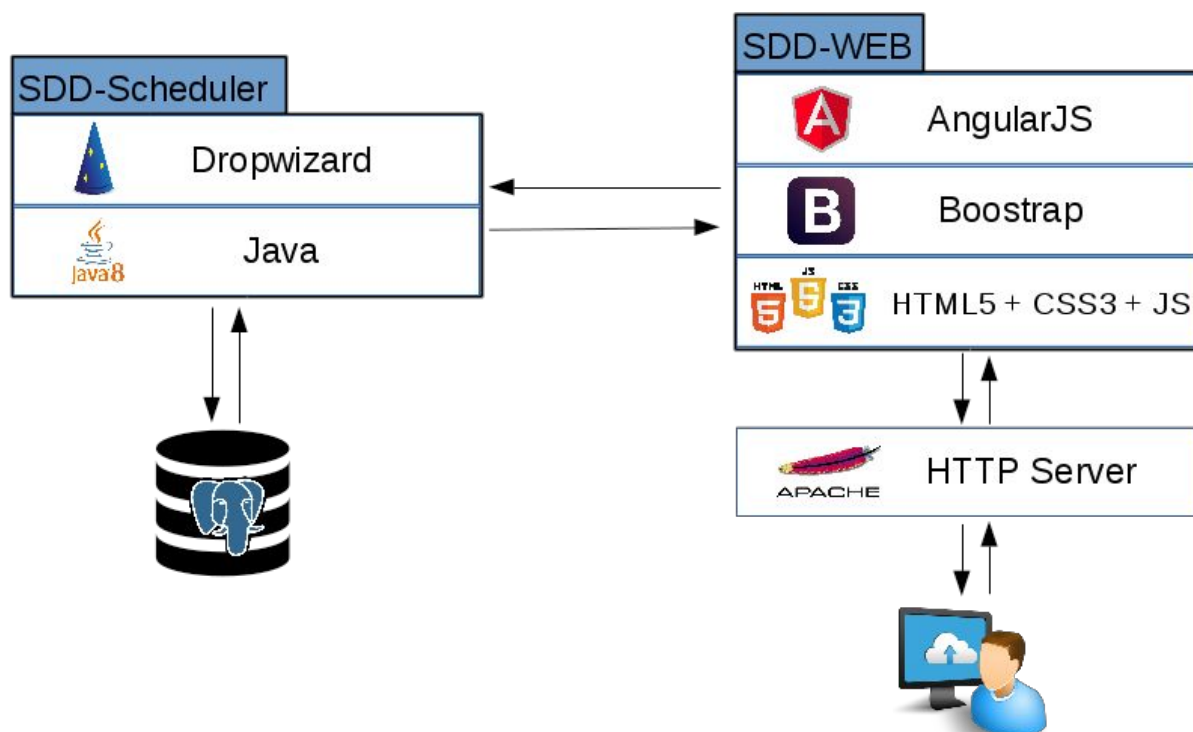


Figura 1: Diagrama da arquitetura do sistema

- **Base de dados:** Base de dados para armazenamento dos dados pertinentes ao sistema utilizando o SGBD postgresql.
- **SDD-Scheduler:** Componente feito em java responsável por controlar os dados do sistema junto com controle de acesso. Ele possui ter uma interface de comunicação utilizando o padrão RESTFUL.
- **SDD-WEB:** Interface web feita para que os usuários interajam com o sistema. Foi desenvolvida utilizando a tecnologia AngularJS e se comunica com o SDD-Scheduler utilizando a sua interface de comunicação RESTFUL.

2. Base de dados

A base de dados do sistema utiliza a ferramenta postgresql na versão 9.1.

Todos os dados e tabelas do sistema são criados de forma automática pelo SDD-Scheduler utilizando o framework **Hibernate** para Java pela sua biblioteca denominada JPA.

O modelo de dados da base de dados do sistema obedece a estrutura que segue abaixo.

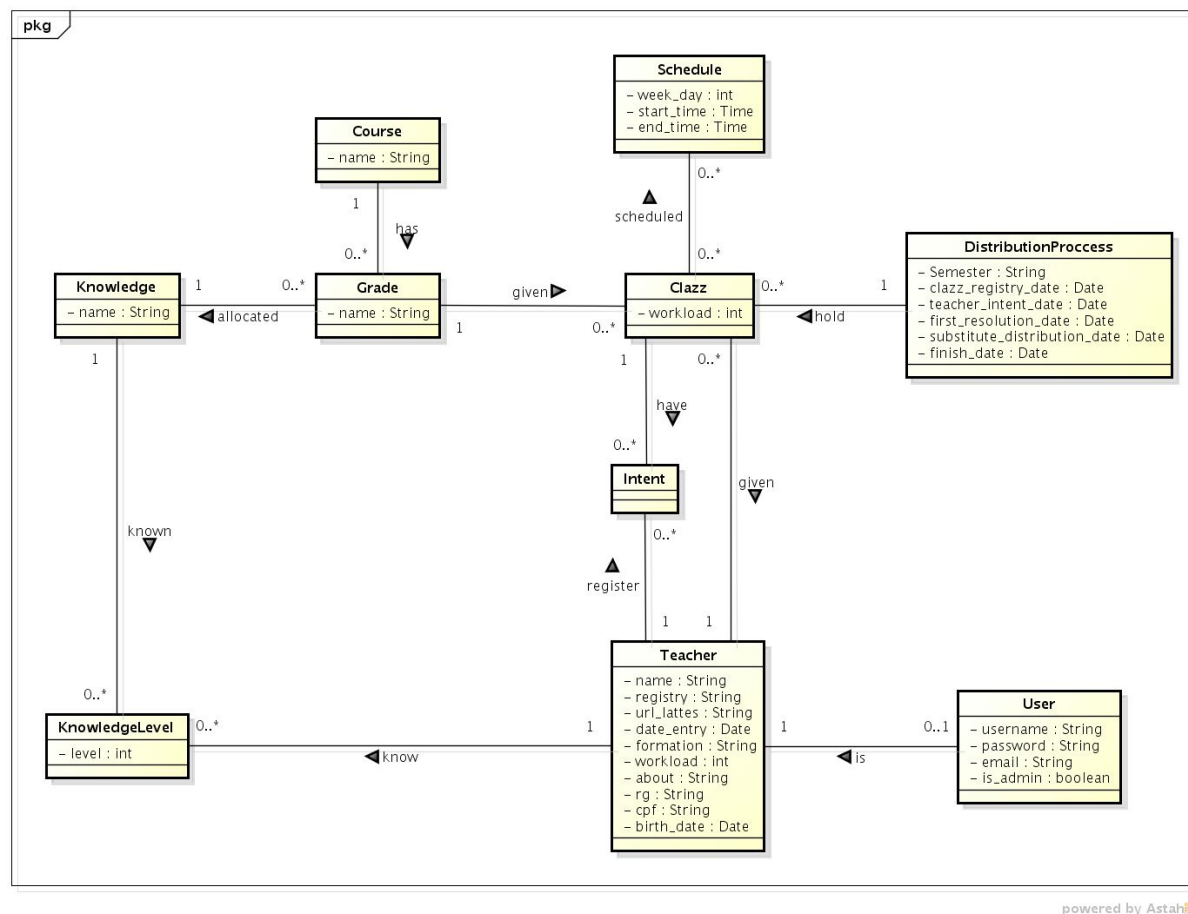


Figura 2: Diagrama de classes do sistema

- User:**
 Entidade responsável por representar um usuário que possui acesso ao sistema, ou seja, uma forma de se autenticar e utilizar o sistema.
 Nesta versão do modelo de dados, espera-se que todo usuário seja um docente, entretanto note que esta abordagem é extensível para que usuários possam ser outras entidades, como por exemplo técnicos administrativos.
- Teacher:**
 Entidade responsável por representar um docente no sistema. Não necessariamente um docente terá acesso ao sistema, para isto, ele deverá possuir um usuário de acesso.
- KnowledgeLevel:**
 Entidade responsável por representar o nível em um determinado núcleo de conhecimento que um professor possui. Basicamente existem 3 níveis:
 - 1 - Apto e deseja ministrar.
 - 2 - Apto mas não deseja ministrar.
 - 3 - Não apto.
- Knowledge**
 Entidade responsável por representar um núcleo de conhecimento.
- Grade:**

SDD-UFG	Versão: 1.0.0
Projeto de arquitetura	Data: 30/11/2015

Entidade responsável por representar uma disciplina de um determinado curso. Uma disciplina pode estar associada a apenas um núcleo de conhecimento pois é a forma que ocorre hoje na distribuição para facilitar a contabilização de horas por núcleo.

- **Course:**
Entidade responsável por representar um curso.
- **Intent:**
Entidade responsável por representar uma inscrição de um docente em uma turma de um processo de distribuição.
- **Clazz:**
Entidade responsável por representar uma turma de uma disciplina que está sendo ofertada em um determinado processo de distribuição.
- **Schedule:**
Entidade responsável por representar horários de aula. A ideia desta entidade é facilitar a geração de relatórios e calendários posteriormente.
- **DistributionProcess:**
Entidade responsável por representar um processo de distribuição. Neste processo serão ofertadas as turmas que serão ministradas pelos docentes no semestre letivo. As datas inclusas nessa entidade definem períodos do processo onde:
 - *clazz_registry_data* até *teacher_intent_date*:
Período de cadastro das turmas que serão ofertadas no semestre letivo pelos coordenadores dos cursos.
 - *teacher_intent_date* até *first_resolution_date*:
Período que os docentes efetivos e os próprios coordenadores dos cursos se inscrevem para ministrar as turmas ofertadas no processo de distribuição.
 - *first_resolution_date* até *substitute_distribution_date*:
Período que o administrador do sistema deve resolver conflitos e entregar as turmas que possuem docentes inscritos para aqueles que irão ministrá-la no semestre letivo.
 - *substitute_distribution_date* até *finish_date*:
Período que o administrado utiliza para alocar as turmas restantes aos docentes substitutos.

3. SDD-Scheduler

O SDD-Scheduler foi desenvolvido utilizando o framework **Dropwizard** (<http://www.dropwizard.io/>), que é um framework desenvolvido em Java para criação de sistemas webservices utilizando o padrão Jetty do Java. Ele implementa um contrato de uma interface de comunicação que utiliza do padrão RESTFUL. O contato está descrito em detalhes, inclusive com acesso a mocks na ferramenta online apiary em <http://docs.sddufg.apiary.io/>.

O componente é o responsável pelo controle de acesso do usuário ao sistema, sendo o responsável pelas sessões dos usuários e o que cada usuário acessou e realizou no sistema, verificando antes de mais nada se este possui permissão para realizar tal operação.

SDD-UFG	Versão: 1.0.0
Projeto de arquitetura	Data: 30/11/2015

O componente também possui a responsabilidade por criar e manter a base de dados utilizando o framework **Hibernate**.

4. SDD-WEB

O SDD-Scheduler é uma interface web que foi desenvolvida utilizando a tecnologia **AngularJS** utilizando do **Bootstrap** framework para construção de sua interface a fim de tentar garantir que seu uso seja responsivo.

O componente deve utilizar do contrato definido para se integrar com o SDD-Scheduler via webservice e a partir daí exibir os dados do sistema e realizar as devidas operações. A ideia do componente é ser uma espécie de “casca” que, utilizando do contrato e dados exportados pelo SDD-Scheduler, ofereça uma forma para que o usuário interaja com o sistema e realize suas devidas operações.

O componente deverá rodar em cima do apache2, um servidor web que possui como objetivo rodar sistemas web.