

Alocação de espaços - NCE	
Documento de Arquitetura	Date: <01/12/2018>

Alocação de Espaços - NCE

Documento de Arquitetura

1. Propósito

Este documento descreve de forma geral a arquitetura do sistema de alocação de espaços, a filosofia que norteia a existência, propósito e evolução do sistema, assim como as qualidades necessárias de código, processos, aptidões dos participantes, entre outras características esperadas dos ambientes de desenvolvimento e de execução.

2. Objetivos da arquitetura e filosofia

O objetivo principal do sistema é permitir acesso amplo de usuários com diversos tipos de dispositivos, com alta disponibilidade durante todos os dias da semana, e levando em conta a diversidade de usuários com necessidades especiais. O acesso à informação é livre para todos, entretanto, somente usuários autenticados e identificados podem alterar o estado do sistema. Além disso, o sistema deve ser simples o suficiente para que todos os usuários possam acessar as informações com o menor número de interações. Finalmente, a manutenção do sistema deve ser barata e exigir poucos recursos humanos, portanto a organização interna do código deve ser simples e direta ao ponto.

3. Suposições e dependências

- O usuário deseja ter acesso ao sistema de alocação de espaços de qualquer dispositivo, a qualquer momento, no geral usará tela de PC padrão e/ou tela de smartphone vertical com taxa de dpi mais alta que a de um PC em geral.
- O ambiente de produção será baseado em nuvem para minimizar os custos operacionais.
- O time de desenvolvimento deve ter experiência nas melhores práticas de desenvolvimento back-end e front-end, e pelo menos um deles com experiência em usabilidade de interface de usuário.
- O software será baseado em web para dar a maior portabilidade, assumindo que o usuário usará tela de PC padrão e tela de smartphone vertical com taxa de dpi mais alta que a de um PC em geral.
- Código simples, com o menor número de tecnologias envolvidas na codificação, usando Java.

Alocação de espaços - NCE	
Documento de Arquitetura	Date: <01/12/2018>

4. Requerimentos significantes da arquitetura

- O front-end será baseado em web: html, css, javascript.
- O back-end será baseado em JavaEE.

5. Decisões, restrições e justificativas

5.1 Interface de usuário web

Interfaces baseadas em web são mais amplamente suportadas pelos mais diversos dispositivos, além de permitirem a estilização para as mais variadas necessidades visuais, como dimensões do dispositivo e também necessidades especiais dos utilizadores.

5.2 Disponibilidade alta

A alocação de espaços deve poder ser feita a qualquer momento, além de poder ser consultada pelos usuários a qualquer momento, de forma a maximizar sua utilidade.

5.3 Linguagem Javascript

Javascript é amplamente difundido e possui grande suporte da comunidade.

5.4 Simplicidade

Recursos para futuros desenvolvimentos do software podem se tornar escassos, tanto na frente financeira como na humana. Portanto o código deve focar na simplicidade para minimizar erros e poder ser corrigido com baixos custos caso algo ocorra de errado.

6. Mecanismos arquiteturais

6.1 Armazenamento de dados

Todo dado recolhido pelo software vindo de usuários deve ser armazenado em um banco de dados central.

6.2 Configuração

Qualquer configuração irá vir de arquivos de configuração, que definem o comportamento geral do software.

6.3 Tratamento de erros

Toda exceção que não for devidamente tratada em código deve ser logada.

6.4 Auditoria

Toda alteração no banco de dados, proveniente de ações do usuário será devidamente logada, com propósito de auditoria.

7. Abstrações principais

7.1 Mapa de ocupação interativo

Serve para visualizar o layout das salas e de forma rápida saber quais estão ocupadas e em que horários. O mapa deve focar na visualização e localização de informações e ainda permitir interação direta para permitir requisitar um slot tempo na sala de acordo com os parâmetros do contexto de visualização.

Alocação de espaços - NCE	
Documento de Arquitetura	Date: <01/12/2018>

7.2 Telas de cadastro em um passo

Toda alteração deve ocorrer com o uso de apenas um passo (sem ter sucessivas telas), retornando ao mapa de ocupação quando terminar.

7.3 Comunicação entre usuários

Um espaço possui responsáveis, assim como, se estiver alocada, possui um usuário que alocou. Cada um desses deve poder ser alcançado para contato, com objetivo de permitir a solução de problemas e obtenção de informações adicionais sobre alocação do espaço.

8. Camadas ou framework arquitetural

O sistema de alocação de espaços no NCE funciona com uma arquitetura cliente-servidor de duas camadas, em que o cliente é o Browser e o servidor é o Apache Tomcat. Há ainda, mais uma camada que fica oculta do ponto de vista do cliente, que se comunica somente com o servidor, que é o SGBD.

8.1 Camada Cliente

A camada cliente se baseia em uma aplicação de browser, programada sobre os recursos do HTML5, que incluem Javascript e as várias APIs disponibilizadas para apresentação e comunicação com o servidor. Não serão feitos post-backs desnecessários, de tal forma a economizar recursos de banda.

8.1.1 Gerenciador de visão

Camada que visa refletir os dados contidos na estrutura de visão. É uma estrutura de dados que contém todas as informações necessárias para renderizar a interface de usuário. Esse gerenciador pode atuar de forma completa ou parcial, caso em que apenas uma parte da interface HTML precisa ser atualizada.

8.1.2 Controle do app cliente

O app cliente possui sua própria lógica de controle, que coordena a interação com usuário na obtenção de dados, salvamento de dados, e estruturação lógica da interface de usuário, tais como dados sendo exibidos, dados temporários de trabalho, entre outros dados da interface. É responsável por ouvir eventos do usuário e tratar tais eventos, tipo salvar algo no servidor, e o contrário também, ouvir eventos do servidor e apresentar ao usuário.

8.1.3 Gerenciador de dados offline

Camada que intermedia todas as operações de obtenção de dados do servidor. Funciona como um cache de dados do tipo write-through, em que os dados escritos vão direto para o servidor, mas os dados lidos são armazenados em uma cache que fica retida no browser, de tal forma que as informações podem ser visualizadas mesmo estando offline.

8.2 Camada Servidora

A camada de servidor se baseia em uma aplicação em JavaEE com apache TomCat. Isso facilita na implementação, evitando erros com uso de dados incorretos para cada função.

8.3 Banco de Dados

O banco de dados será inicialmente composto por arquivos de texto, mas depois podem ser usados SGBD que são mais robustos, e permitem operações feitas de forma segura.

9. Visões arquiteturais

9.1 Lógica

Módulo de interface de usuário

Módulo servidor

Separação em model, view e controller em cada uma das camadas (i.e. cliente e servidor)

Alocação de espaços - NCE	
Documento de Arquitetura	Date: <01/12/2018>

9.2 Casos de uso

Convidado: usuário que deseja obter informações sobre a alocação de espaços.

Requerente: usuário que deseja alocar para si um espaço e horário.

Responsável: usuário gerencia a alocação de um ou mais espaços que estejam sob sua responsabilidade.