



UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA



Autorizada pelo Decreto Federal nº 77.496 de 27/04/76
Recredenciamento pelo Decreto nº 17.228 de 25/11/2016
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
COORDENAÇÃO DE INICIAÇÃO CIENTÍFICA



Fapesb
Fundação de Amparo
à Pesquisa do Estado da Bahia

XXV SEMINÁRIO DE INICIAÇÃO CIENTÍFICA DA UEFS SEMANA NACIONAL DE CIÊNCIA E TECNOLOGIA - 2022

Desenvolvimento do Back-end do “Portal Acervus cultura escrita e memória literária Parte II”

Mateus Lima Fonseca da Conceição¹; Patrício Nunes Barreiros²

1. Bolsista PIBIC/CNPq, Graduando em Engenharia da computação, Universidade Estadual de Feira de Santana, e-mail: lima.mt96@gmail.com
2. Orientador, Departamento de Letras, Universidade Estadual de Feira de Santana, e-mail: patricio@uefs.br

PALAVRAS-CHAVE: Filologia; Crítica Textual; Eulálio Motta

INTRODUÇÃO

A pesquisa propôs a conclusão do back-end e a realização dos testes de funções do *Portal Acervus: cultura escrita e memória literária*. As atividades foram iniciadas anteriormente (plano de trabalho PIBIC 2020-2021), mas a criação e implementação de algumas funções como o banco de dados precisou de mais uma etapa para ser implementado e testado. O objetivo do Portal é oferecer aos usuários o acesso a documentos de acervos de escritores e edições filológicas elaboradas pelo grupo de pesquisadores do Núcleo de Estudos Interdisciplinares em Humanidades Digitais. O teste piloto do acervo foi feito a partir da obra de Eulálio Motta.

O back-end corresponde a todas as atividades relacionadas ao funcionamento do site, por detrás da interface visível pelo usuário. O desenvolvedor do back-end é responsável pela codificação do site, pelo banco de dados e a programação dos retornos que serão visualizados pelos usuários do site, portanto, o back-end é a alma de uma aplicação web. O plano de trabalho “Desenvolvimento do back-end do Portal Acervus: cultura escrita e memória literária (Etapa 02)” faz parte do projeto de pesquisa “Edição das obras inéditas de Eulálio Motta (IV etapa)”, que tem o objetivo de elaborar edições filológicas impressas e digitais das obras do escritor baiano, a partir dos manuscritos preservados no acervo do escritor.

A pesquisa propõe uma aproximação efetiva entre a Filologia e a Informática, com o intuito de desenvolver um sistema que possa disponibilizar as edições digitais elaboradas pelo grupo de pesquisadores do projeto “Edição das obras de Eulálio Motta” e otimizar a preparação de edições digitais, realizar anotação lexical, gerar glossários e vocabulários, e preparar corpus para estudo linguístico.

Como se pode notar, a partir da descrição acima, o “Portal acervus: cultura escrita e memória literária” não é um site comum, trata-se de um sistema com banco de dados robusto, com funções complexas de armazenamento e recuperação de dados, sistema de administração e um emaranhado de telas, exigindo dos desenvolvedores conhecimentos de linguagem de programação, banco de dados e arquitetura de sistemas. O desenvolvedor do back-end tem a responsabilidade de criar o sistema de segurança dos dados e de manutenção do sistema. Para a elaboração do portal serão seguidos os protocolos de requisitos elaborados por Barreiros (2019), com o intuito de orientar o trabalho dos programadores envolvidos no projeto. Esse conjunto de instruções foram descritas num protótipo que contém a descrição de todas as telas e suas funções.

A pesquisa baseia-se numa abordagem interdisciplinar que converge conhecimentos de filologia e informática, centrando-se num campo de estudo denominado de Humanidades Digitais. Para Barreiros (2018), as Humanidades Digitais têm como principal característica aproximar as ciências humanas da informática, promovendo um diálogo multidisciplinar e colaborativo capaz criar novas metodologias de trabalho, a partir da erudição inerente às humanidades e à capacidade da informática de criar aplicações para resolver questões pontuais do cotidiano.

De acordo com Barreiros (2019), a Filologia sempre manteve um diálogo profícuo com disciplinas consideradas afins e, ao longo de sua história, lançou mão das inovações tecnológicas relacionadas aos mecanismos de produção, difusão e apropriação dos textos. Isso a aproximou da Paleografia, da Epigrafia, da Papirologia, da Diplomática e da Bibliografia Textual, por exemplo. Atualmente, a cultura escrita adentrou numa nova fase – a cultura digital, que tem uma lógica diferente das culturas manuscrita e impressa. Portanto, é natural que os filólogos comecem a repensar as suas práticas, com o objetivo de compreender o texto fora de uma superfície física e encontrar novos formatos de edições. Sabe-se que a escrita é uma das mais importantes tecnologias inventadas pelo homem e, nas sociedades letradas, ela ocupa um papel central, sendo responsável pelo equilíbrio e manutenção do organismo social.

Aliada aos computadores, a escrita passou a fazer parte de algo mais amplo, as chamadas Tecnologias da Informação e da Comunicação em Rede, que têm contribuído para o surgimento de uma sociedade baseada em práticas cotidianas dependentes dos novos aparatos tecnológicos e dos novos sistemas de comunicação. É com o intuito de explorar novos caminhos para a prática editorial, no âmbito da Filologia, que propomos a criação de um sistema computacional, na expectativa de atender aos anseios da comunidade científica e dos leitores/usuários de um modo geral. Nesse sentido, a presente pesquisa atua no âmbito da inovação, abrindo espaço para novas possibilidades de conceber as edições filológicas e sua difusão, democratizando o acesso dos conhecimentos produzidos na universidade.

METODOLOGIA

A pesquisa segue a metodologia empregada nas edições digitais (hiperedições) das obras de Eulálio Motta, desenvolvida por Barreiros (2018, 2015, 2013), na documentação e no protótipo para a elaboração do “Portal acervus: cultura escrita e memória literária” (BARREIROS, 2019); e nos critérios e princípios postulados pela Modern Language Association (SHILLINGSBURG, 2018).

No âmbito da computação, a pesquisa lida com sistemas capazes de desenvolver bancos de dados e integrá-los às interfaces da aplicação em desenvolvimento.

Materiais

- Visual Code Studio
- Node.JS
- Framework Express
- Desenvolvedor de API Insomnia

Estudo e domínio da linguagem de banco de dados:

Partindo do conceito de banco de dados, foi escolhida como linguagem a *JavaScript* ambientado em *Node.js* que se define como um ambiente de execução *JavaScript server-side*. Isso significa que com o *Node.js*, os aplicativos *JavaScript* podem ser criados para rodar a aplicação de modo independente na máquina, não dependendo do navegador para rodar, como ocorre na forma usual (BROWN, 2010). Para os requisitos funcionais

do Portal Acervus cultura escrita e memória literária, o *framework Express.js* foi utilizado para desenvolver o banco de dados.

Levantamento de Requisitos:

O levantamento de requisitos necessário para estruturação da base do banco de dados resultou num diagrama de classes. O diagrama é a primeira abstração de um modelo materializado e ilustra graficamente como será a estrutura do banco de dados. No diagrama de classes temos a especificação dos componentes e as funções designadas a cada item, como a exemplo do modelo de biografia que detêm objetos como cronologia e poeta, e as funções de acesso à determinada biografia e acesso ao determinado poeta.

Modelagem e prototipagem de banco de dados:

A modelagem e prototipagem de banco de dados foram criadas utilizando Unified Modeling Language (UML). Como o nome indica, é uma linguagem de notação utilizada para modelar e documentar as diversas fases do desenvolvimento de sistemas orientados a objetos (LARMAN, 2000), inclusive modelagem de banco de dados. O tipo de diagrama utilizado foi o diagrama de classe, o mesmo mapeia de forma clara a estrutura de um determinado sistema ao modelar suas classes, atributos, operações e relações ente objetos.

Acessos (Perfis dos usuários), conteúdo, Gerenciamento de Conteúdo, busca, mapa do site, segurança, hospedagem:

Para os acessos do Portal Acervus, também baseados no diagrama de classes, foi construído o sistema de login e senha para cada usuário e suas respectivas hierarquias, sendo elas administrador, usuário comum, e colaborador. E pensando na segurança do usuário, as senhas são criptografadas em MD5, gerando um token, que permite o uso do sistema através de uma ou mais sessões.

A busca de conteúdo foi baseada em Create, Read, Update, Delete (CRUD), esse sistema é de extrema necessidade para todo e qualquer componente de banco de dados, são funcionais de criação, leitura, atualização e exclusão de arquivos.

O mapa do site em função do back-end foi também baseado pelo diagrama de classes e todos os modelos e controllers, foram devidamente implementados com as funções básicas CRUD.

A segurança do acervo é tratada em sua hospedagem, Hostinger, que há um sistema o Patchstack que repele malwares e usuários maliciosos reforçando a segurança de dados.

RESULTADOS E DISCUSSÃO

Como resultado da pesquisa foi possível desenvolver o banco de dados (back-end) estrutural para o portal *Portal Acervus cultura escrita e memória literária*. A linguagem *Java script* ambientado em Node.js mostrou-se adequada para a finalidade proposta. Os requisitos funcionais do Portal foram desenvolvidos satisfatoriamente no *framework Express.js*.

As estratégias utilizadas para o levantamento de requisitos foram adequadas para o projeto, possibilitando visualizar como a aplicação funcionaria. O diagrama de classe permitiu mapeara de forma clara como o sistema funcionaria e como as relações seriam estabelecidas.

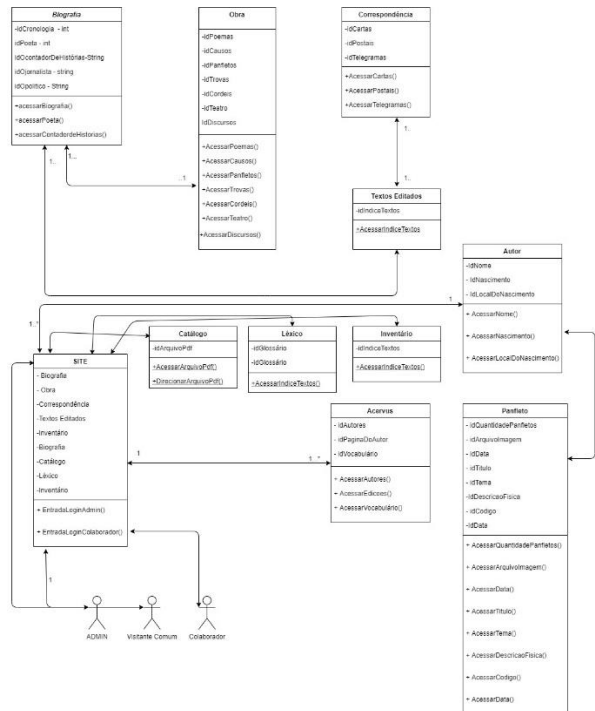


Figura 1: Diagrama de Classe modelo (UML)

Os testes para o login na ferramenta Insomnia (Figura 2), mostrou-se adequado para dialogar com a interface do sistema, permitindo responder aos que se espera das funções.

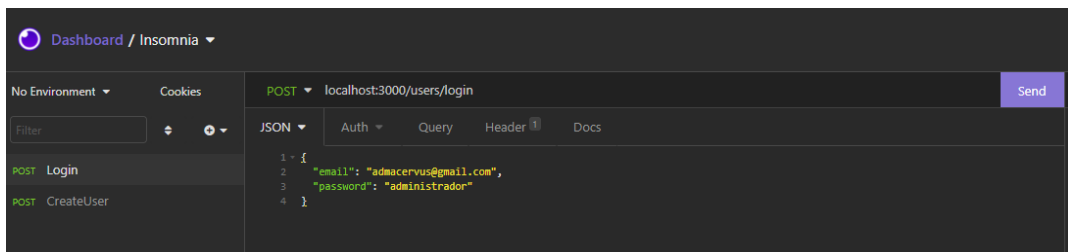


Figura 2: Ambiente de criptografia de usuários, login e senha Insomnia.

A busca de conteúdo baseada em Create, Read, Update, Delete, atendeu ao que se esperava para os componentes do banco de dados. Os testes de segurança atenderam ao que foi requisitado. Portanto, a metodologia empregada foi suficiente para entregar um produto como requisitado pelo cliente.

CONCLUSÃO

Conclui-se com o desenvolvimento do Back-end do Portal Acervus cultura escrita e memória literária que a pesquisa de acervos e bibliotecas similares contribuiu para nortear como o desenvolvimento do projeto. O estudo e o domínio da linguagem escolhida adaptaram-se à acessibilidade do projeto e sua flexibilidade de manipulação de arquivos e dados. Seu levantamento de requisitos possibilitou a elaboração de um diagrama de classe que se tornou a base da modelação e prototipação do banco de dados, que será utilizado para abrigar o Portal Acervus.

O servidor ativo e online, com o uso do mysql juntamente com o node.js, a API desenvolvida permitiu o gerenciamento das informações dos autores presentes no portal Acervus, sendo possível obter informações biográficas e o acesso às edições digitais. Por

fim suas finalidades, funções, requisitos de segurança hospedagem e segurança do banco de dados, finalizaram essa etapa do projeto.

REFERÊNCIAS

BARREIROS, Patrício Nunes. *A Prototipagem do Portal Acervus: Cultura Escrita e Memória Literária*. Disponível em: www.neihd.wordpress.com/bibliografia. Acesso em: 02 de setembro de 2022.

BARREIROS, Patrício Nunes. A relevância do dossiê arquivístico em edições digitais de documentos de acervos de escritores. *Revista Internacional del Libro, Digitalización y Bibliotecas*, v. 2, p. 20-33, 2014.

BARREIROS, Patrício Nunes. Novas práticas culturais da escrita, novas perspectivas da Crítica Textual: rumo às hiperedições. *Linguística e Filologia Portuguesa (USP)*, São Paulo, v. 16, 2014.

BROWN, Ethan. *Programação web com Node e Express: Beneficiando-se da stack JavaScript*. Santa Terezinha (SP): Novatec Editora, 2020.

LARMAN, Craig. *Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos*. Porto Alegre (RS): Bookman, 2000.

SHILLINGSBURG, Peter L. *Scholarly editing in the computer age: theory and practice*. 3. ed. Michigan: University Michigan, 2004.