|  |  |  |
| --- | --- | --- |
|  | **Ministério da Educação**  **Universidade Tecnológica Federal do Paraná**  ***Diretoria De Graduação e Educação Profissional***  ***Coordenação do Curso Técnico Integrado em Informática*** |  |

FORMULÁRIO DE AVALIAÇÃO PERIÓDICA DO ESTÁGIO

Nome do Estagiário: João Martins de Queiroz Filho RA: 1552260

Nome da Empresa: Desenvolvimento de uma plataforma web para suporte a cursos de computação paralela

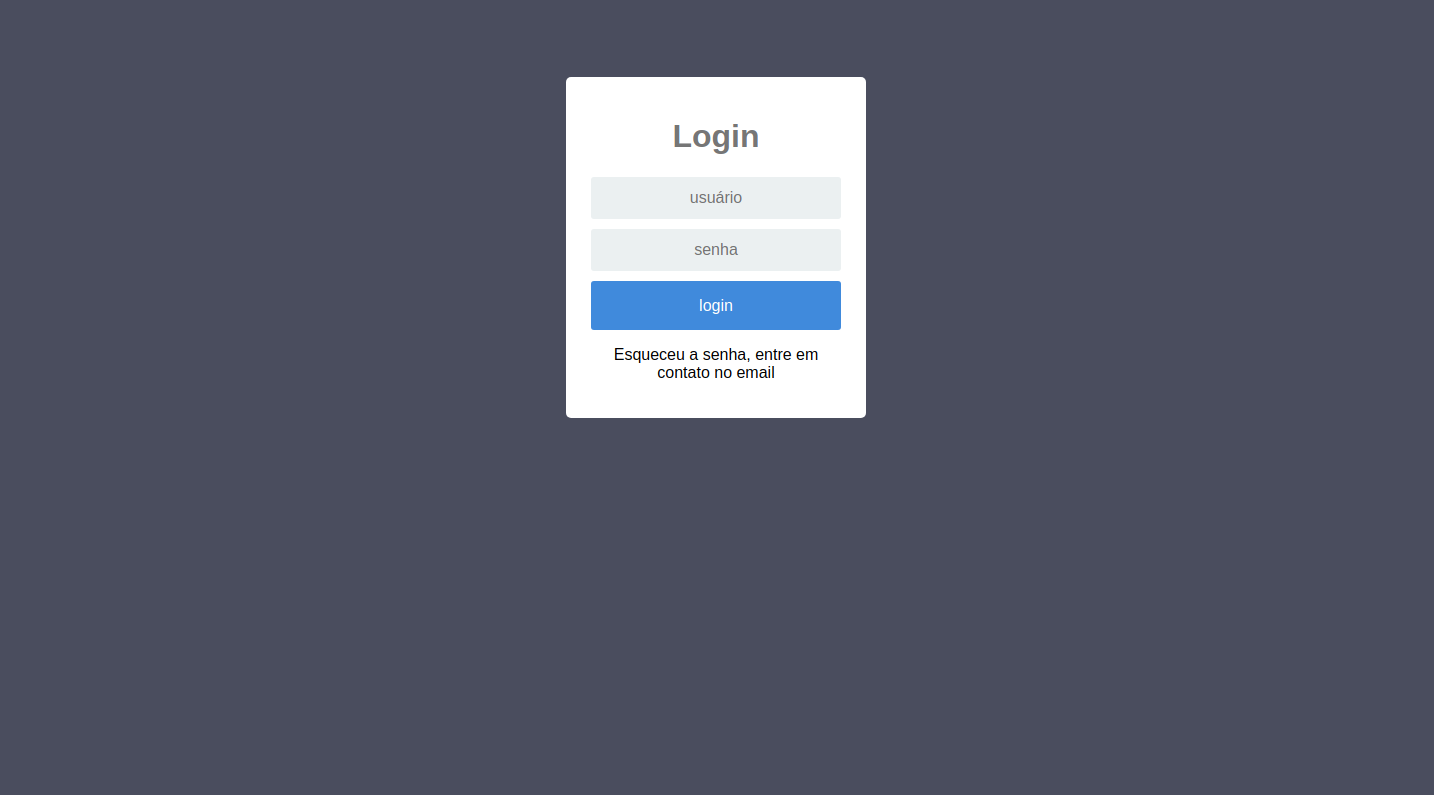
Nome do Supervisor (Empresa): Rogério Aparecido Gonçalves

Nome do Orientador (UTFPR): Marcos Silvano Almeida

Período avaliado: de 14/05/2018 a 28/05/2018

Atividades Desenvolvidas (inclua as dificuldades, os conhecimentos e experiências adquiridas no período):

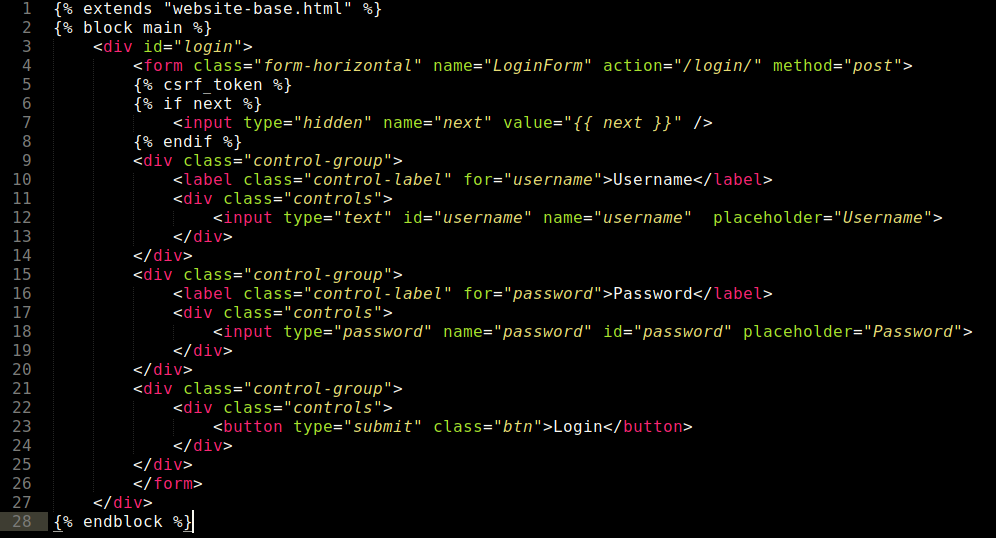
Com a criação da tabela de dados contendo os dados dos alunos, foi criado em *HTML* uma tela de login para o aluno, na qual ele realizará a autenticação para resolver os exercícios propostos pela plataforma. Pode ser visualizado na Figura 1 a tela de *login.*

  
Figura 1: Tela de login visto pelo usuário

Para a criação da tela de desenvolvimento, bem como as outras telas da plataforma foi utilizado o *framework BootStrap* que facilita o desenvolvimento do *front-end.* Esse *framework* utiliza da linguagem *HTML e CSS* para o seu desenvolvimento.

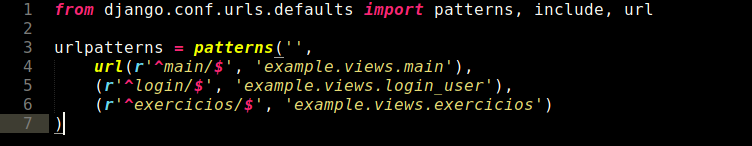
Com o desenvolvimento da tela de *login,* partiu-se para o desenvolvimento do *back-end* que é o local que realiza as autenticações do usuário. Como nunca havia trabalhado com *Django* houve também a dificuldade de criar essa autenticação.

Explicado nos outros relatórios o *Django* trabalha com o modelo *MVC,* onde cada parte da plataforma fica no devido local, por exemplo as tabelas são criadas no pacote *model,* todos os *templates* usados, isto é, todas as janelas que serão visíveis para o usuário são colocados neste pacote, por fim, o pacote de controle que é feito para gerir todos os acessos e autenticações, em *Django* é chamado de *urls.* Pode ser visto na Figura 2, como foi criado a tela de *login* em *HTMl.*

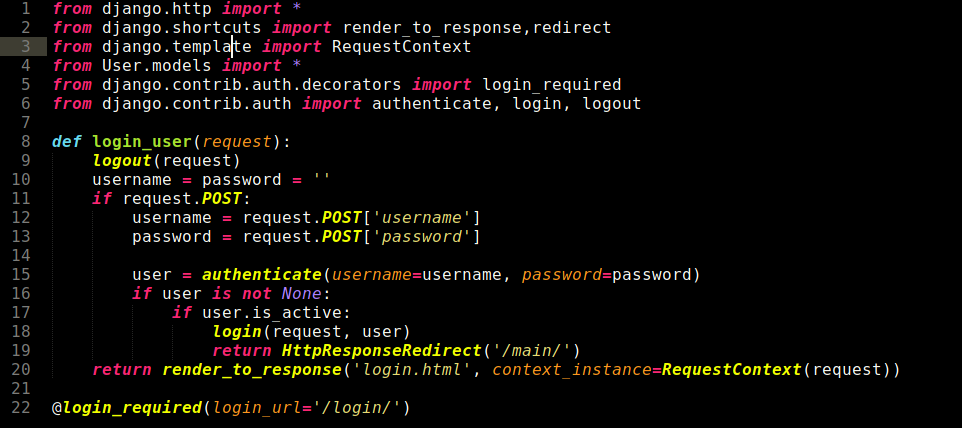
  
Figura 2: Código realizado para a tela de Login

Como o *Django* trabalha com a linguagem de programação Python, todos os métodos *GET e POST* devem ser feitos em *Python* o que ocorreu uma outra dificuldade e um tempo gasto para aprendizado. Primeiro nas linhas 1-2, é criado um *annotation,* isto é, é especificado que aquela parte do código é para ser executado da seguinte forma. Em *Python* o método POST é feito especificando primeiro na *class* do form e logo após é criado variáveis que realizarão o envio e recebimento dos valores, este valor enviado é especificado na linha 7 - *{{ next }}.*

Criado a tela de *login* no *view*, deve-se especificar as *urls,* isto é, o *controller,* onde fica especificado qual o caminho que se deve realizar após o acesso do usuário. Essa especificação é feita por expressão regular, o código é mostrado na Figura 3.

  
Figura 3: Especificação do caminho do usuário no arquivo urls.py

Por fim, é implementado a *view* que fica responsável para a requisição com banco de dados e autenticação do usuário. Na Figura 4 pode ser visualizado a implementação da *view.*

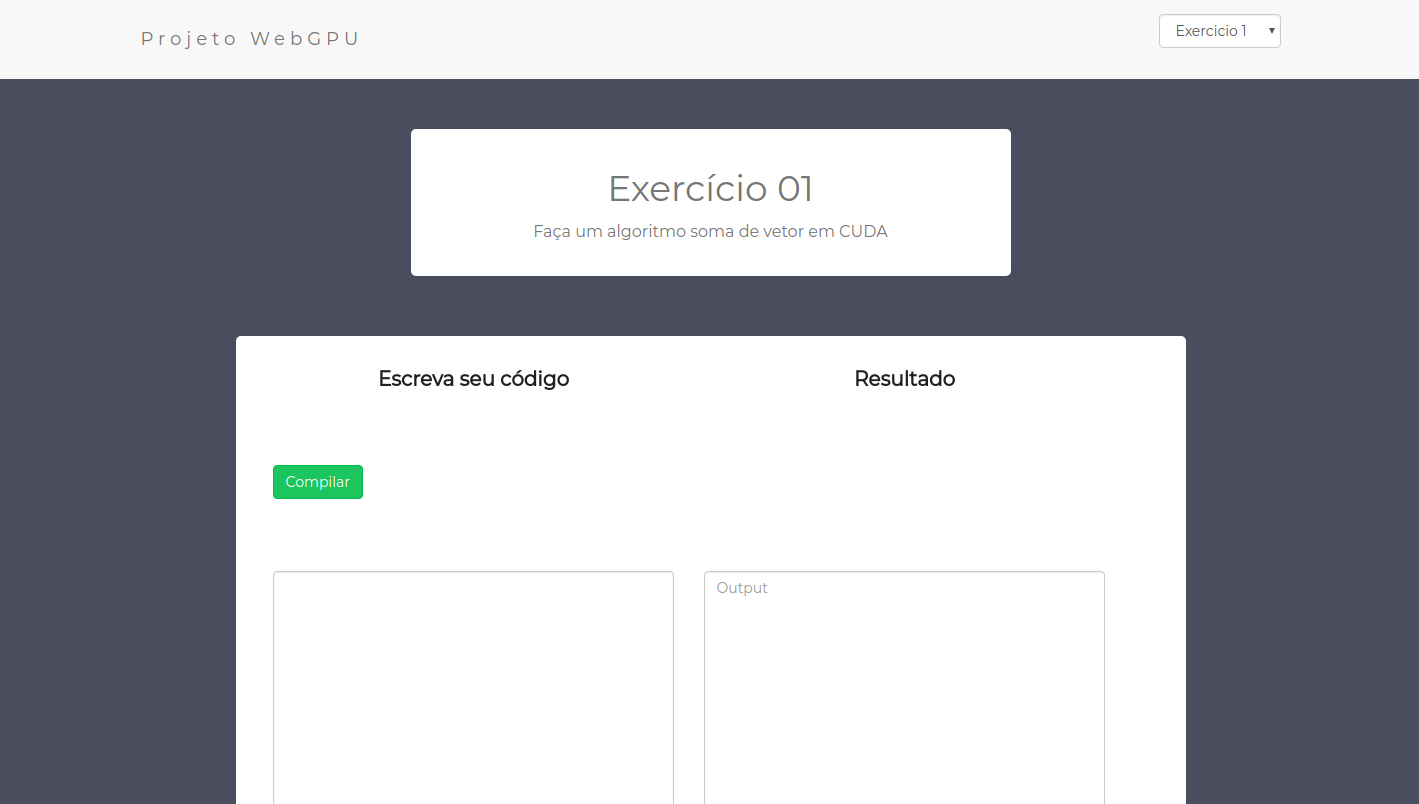
  
Figura 4: Implementação da view para o login do aluno

Para o desenvolvimento da *view* hoje também um tempo para o aprendizado do funcionamento do método *authenticate,* bem como entender os métodos *GET e POST* em *Python,* pois até o momento tinha feito esses métodos em *PHP*, que é muito diferente. Contudo com o aprendizado a implementação se torna rápida e tranquila de se fazer, sendo até melhor que o desenvolvimento em *PHP.*

Uma outra dificuldade interessante é que os arquivos *HTML* não funciona os códigos em *CSS* se não estiver dentro do mesmo arquivo .*html*, para resolver o problema foram criados os *CSS* dentro do arquivo *.html,* contudo, o código fica desorganizado e poluído.

A partir da autenticação do usuário, foi criado a tela de exercícios, onde o aluno estara desenvolvendo os algoritmos e também realizando a compilação nas placas de vídeo que estarão instanciadas para a plataforma.

A implementação da tela de exercícios é comparável a tela de login, contudo sua diferença é que o aluno desenvolverá o código na plataforma, quando ele clicar no botão **Compilar,** este salvará o código desenvolvido na base de dados, para que haja um controle e estatística por parte do professor e depois requisitará à placa de vídeo o resultado da execução do código. Quando o aluno verificar que está correto seu código, ele pode ir para os próximos exercícios que estarão no conto direito da tela Na Figura 5 é mostrado a tela de exercícios.

  
Figura 5: Tela de exercícios onde o aluno poderá desenvolver seus códigos CUDA

Outra dificuldade encontrada nesta parte do desenvolvimento foi entender como que havia o controle de concorrência por parte do *Python* na utilização da placa de vídeo. Até o momento está um pouco abstrato o seu funcionamento, para realizar a chamada para a placa de vídeo é usado a biblioteca *sys* que não tem nenhuma documentação que explica como que esse controle funciona.

Também houve a dificuldade no conceito do aluno desenvolver o código na plataforma, pois para enviar para a placa de vídeo deve ter um ajuste no código, a primeiro momento o ajuste foi gerar um arquivo no formato *.cu (*da linguagem CUDA)nos diretórios da máquina que está instanciada a placa de vídeo e esta faz a compilação e execução.

A plataforma ainda não esta em funcionando, para instanciar deve-se realizar modificações na biblioteca CUDA e também atualização do sistema operacional (Debian), estas modificações e atualizações serão feitas após o uso das máquinas para o Trabalho de Conclusão de Curso do aluno João Martins.

A partir de instanciada será realizado modificações e manutenções para que a plataforma fique apto a ter alunos e ser usado para o minicursos e projetos voltado a educação para desenvolvimento de código para plataformas heterogêneas.

Observações: Esse formulário deverá ser preenchido periodicamente (a cada 2 semanas de estágio) e assinado pelos responsáveis indicados abaixo. Arquive-os e, ao final do estágio, todos os formulários deverão ser anexados ao Relatório Final de estágio.

Assinaturas:

|  |  |  |
| --- | --- | --- |
| Aluno \_\_\_/\_\_\_/\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Supervisor \_\_\_/\_\_\_/\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Orientador \_\_\_/\_\_\_/\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Autor: Lucio Valentin