|  |  |  |
| --- | --- | --- |
| **Curso:** ENGENHARIA DE COMPUTAÇÃO | **Ano: 2015** | **Semestre: 2015-1** |
|  | | |
| |  | | --- | | **Estudante 1: Andrew Kretzer**  **Estudante 2: Luis Ricardo Sabino**  **Estudante 3: Matheus Engleitner** | | **Estudante 4: Felipe Ferreira** | | **Estudante 5: Thaian Andreia da Silva**  **Estudante 6: Ramon Amorin** | | | |
| **Tema/ Título: SmartHouse – Sistema interativo para casa inteligente** | | |
|  | | |
| **JUSTIFICATIVA E PROBLEMATIZAÇÃO**  Devido a atual situação econômica e ambiental do país, na qual temos um elevado custo mensal com tributações e a escassez de recursos naturais em diversas regiões, o consumo sustentável tornou-se um tema bastante abordado no cotidiano dos brasileiros.  Tendo como uma alternativa na redução dos gastos, a automatização residencial, que possibilita o controle de gastos e automatização de processos residenciais, proporciona ao seu usuário benefícios econômicos, conforto e agilidade em algumas tarefas do dia-a-dia, além de contribuir com questões ambientais, como a redução do consumo dos recursos naturais, através do controle e identificação de vazamentos de água ou equipamentos defeituosos na rede elétrica.  Pensando nos pontos citados anteriormente, fica a dúvida de como criar uma forma de resolver esses problemas. Será que um sistema de automatização residencial seria uma boa solução? Quais ferramentas esse sistema deveria ter e como seria a utilização dessas ferramentas pelo usuário que o adquirir?  A partir dessas perguntas, será desenvolvido o projeto SmartHouse, que une os benefícios da automatização residencial a um software de monitoramento interativo, com diversos recursos, que trará ao seu usuário uma forma simples de acompanhar seu consumo residencial remotamente, além de realizar diversas funções que trarão economia e conforto. | | |

|  |
| --- |
| **Objetivo Geral:**  Explorar as formas de desenvolvimento e comunicação entre sistemas embarcados e uma plataforma Web para a criação de um sistema de controle, interação e automatização residencial. |
| **Objetivos Específicos:**  * Definir ambiente de desenvolvimento e as tecnologias que serão utilizadas. * Criar um ambiente para desenvolvimento em equipe. * Estudar a comunicação entre microcontroladores e o sistema web por meio de webservices em uma rede ethernet. * Estudar as linguagens e tecnologias que serão aplicadas ao desenvolvimento. * Modelar sistema proposto de acordo com as especificações dos requisitos e descrição dos processos do sistema. * Desenvolver o projeto de acordo com os objetivos específicos citados anteriormente e requisitos do sistema. * Criar um ambiente de comunicação local, por meio de switch e cabos ethernet. * Realizar a comunicação entre os microcontroladores e o sistema. * Definir telas, módulos e rotinas do sistema de acordo com os requisitos. * Criar as telas, programas e funcionalidades requisitadas ao sistema. * Elaborar e realizar testes para a solução de possíveis inconsistências. * Corrigir inconsistências encontradas na etapa de testes. * Criar apresentação do projeto, com as funcionalidades que ele integra. |

|  |
| --- |
| **Desenvolvimento do Trabalho:**  Definição de tecnologias:  **Desenvolvimento:** ASP.Net, HTML5, SQL, JQuery, Javascript e CSS.  **Armazenamento de Dados:** SQL Server 2012.  **Framework:** .Net Framework 4.5, Bootstrap Admin LTE.  **Transmissão de dados:** protocolo de rede TPC/IP, comunicação via cabos ETHERNET, envio e recebimento de dados por variáveis de texto via WebService. Servidor web IIS, a aplicação rodará de forma local para garantir que funcione corretamente durante a apresentação, não havendo assim a necessidade de conexão via internet.  **Plataforma Operacional:** Sistema Operacional Microsoft Windows 7, plataforma de desenvolvimento Microsoft Visual Studio 2013. |

|  |
| --- |
| **Cronograma de Atividades**  **Março:** Definição das tecnologias  Entrega do pré-projeto  Pesquisa bibliográfica  Início do desenvolvimento do projeto  **Abril:** Desenvolvimento do sistema  Documentação da metodologia aplicada  Criação dos modelos do projeto  **Maio:** Desenvolvimento do sistema  Documentação da metodologia aplicada  Criação dos modelos do projeto  Conclusão e entrega dos requisitos  Apresentação do projeto  **Maio:** Revisão.  Entrega do projeto Final.  **REQUISITOS**  **REQUISITOS FUNCIONAIS**   1. **Consultas**   **Descrição:** O sistema deve fornecer ao usuário diversos indicadores dos dados que serão coletados, para que o usuário possa acompanhar os gastos residenciais. Esses indicadores serão gerados a partir dos dados de consumo que serão enviados para o sistema a partir dos microcontroladores, eles serão um resumo do consumo de água, energia elétrica, além dos dados coletados pelos demais sensores.  **Prioridade:** Alta   1. **Interação**   **Descrição:** O sistema deve fornecer ao usuário formas de interagir com os dispositivos físicos do projeto por meio de comandos do sistema, esses comandos serão componentes em tela que possibilitem a entrada de valores que serão enviados aos microcontroladores para realizar uma tarefa específica como ligar ou desligar equipamentos residenciais.  **Prioridade:** Essencial   1. **Relatórios**   **Descrição:** O sistema deverá realizar a emissão de relatórios de consumo a partir dos dados gravados na base.  **Prioridade:** Alta  **REQUISITOS NÃO FUNCIONAIS**   1. **Ambiente de desenvolvimento**   **Descrição:** O ambiente de desenvolvimento deverá ser escolhido de acordo com as necessidades no sistema e conhecimento dos integrantes da equipe.  **Prioridade:** Alta   1. **Comunicação**   **Descrição:** O sistema deverá se comunicar com os dispositivos Arduíno de forma que seja possível o envio e recebimento dos dados.  **Prioridade:** Alta   1. **Acessibilidade**   **Descrição:** O sistema deve permitir que o usuário o acesse remotamente, via internet, realizando as consultas e interações de qualquer lugar  **Prioridade:** Alta   1. **Segurança**   **Descrição:** O sistema deve garantir que não haja interferência de terceiros nas informações coletadas, através de criptografia de dados.  **Prioridade:** Alta   1. **Interface**   **Descrição:** A interface do sistema deverá ser de fácil entendimento ao usuário  **Prioridade:** Alta   1. **Responsividade**   **Descrição:** O sistema deve possuir recursos responsivos, ou seja, se adaptar de acordo com a tela do dispositivo de acesso.  **Prioridade:** Alta   1. **Base de Dados**   **Descrição:** O sistema deverá guardar as informações de consumo colhidas pelos microprocessadores.  **Prioridade:** Alta   1. **Modelos de Visualização**   **Descrição:** O projeto do sistema deverá apresentar os três modelos PIM (Casos de Uso, Modelo Hierárquico de requisitos e Modelo) e os dois modelos PSM (Modelo de Bloco de e Modelo de Bloco Interno).  **Prioridade:** Alta   1. **Refinamento de consultas**   **Descrição:** Os relatórios serão emitidos de acordo com os valores dos filtros que serão disponibilizados ao usuário para melhor refinamento das consultas.  **Prioridade:** Alta   1. **Rede**   **Descrição:** Será necessária a criação de uma rede local que irá fazer a conexão entre o servidor da aplicação e os microprocessadores.  **Prioridade:** Alta   1. **Apresentação**   **Descrição:** Ao final do desenvolvimento, deverá ser realizada uma apresentação em forma de vídeo ou slides sobre o sistema, seu funcionamento e especificações.  **Prioridade:** Alta  **METODOLOGIA**  **Ambiente de trabalho**  Para o desenvolvimento do sistema, é necessário realizar algumas configurações para que seja possível a comunicação entre um projeto rodando numa máquina local e dispositivos físicos que irão enviar e receber informações.  A comunicação é feita através do IIS (serviços de informação de internet), o projeto ASP.NET que será desenvolvido terá um site local no IIS, a configuração é feita através de um IP fixo atribuído à máquina em que o projeto está instalado e uma porta específica.  Deve-se desabilitar o firewall da rede local para que a comunicação entre os periféricos seja permitida, na plataforma de desenvolvimento, a URL de execução do projeto será a mesma do site criado no IIS.  Após realizar corretamente estas etapas, o projeto estará acessível em uma rede local e o desenvolvimento será iniciado.  **Banco de Dados**  Os dados do sistema serão armazenados em um banco de dados SQL Server, para isso, será utilizado o SQL Server Instalation Center, está ferramenta permite a criação de instâncias de banco de dados e instalação de programas e serviços disponíveis do SQL Server.  Para o projeto, além da criação da instância e serviços necessários para o funcionamento da mesma, será utilizado o SQL Server 2014 Management Studio para gerenciamento do banco de dados, tabelas e registros.  **Projeto SmartHouse**  Conforme os requisitos, o Web Site do projeto será desenvolvido em ASP.NET, nas linguagens ASPX para a camada visual e C# para a camada lógica.  **Criação:**  No Visual Studio, será criado uma nova solução (Solution, arquivo .sln) em branco, nomeada Smarthouse, conforme especificado no blog oficial da Microsoft, as soluções são estruturas para a organização de projetos no Visual Studio, no caso do sistema SmartHouse, a solução terá dois projetos, o projeto App, criado para a camada visual e o projeto Lib, onde estarão as bibliotecas, classes, funções e lógica do sistema.  A arquitetura do sistema será baseada no padrão MVC (Model View Controller), que, basicamente, um padrão moderno para aplicações Web, nele, as camadas separadas trazem maior organização ao projeto, facilitando a modularização e manutenção do código. Esta escolha foi feita devido ao grande número de necessidades que o sistema poderá abranger futuramente e que, com o uso dessa arquitetura, a implementação de novas funcionalidades será mais fácil e organizada.  **Desenvolvimento do Web Site (Projeto App)**  Inicialmente, será adicionado ao projeto o template Admin LTE, um Framework para front-end, baseado no Twitter Bootstrap 3, uma coleção de ferramentas de código aberto para a criação de websites e aplicações web. O template Admini LTE foi escolhido devido às suas funcionalidades de Dashboart e formulários que serão necessários à interface do projeto.  No projeto App será criada uma pasta (assets) para adicionar todos os arquivos do template.  O ASP.Net possibilita a criação de Master Pages, estas funcionam como um tipo de página mestra à todas as demais páginas, abstraindo o código que é comum à todas elas. No caso do projeto, na Master Page será linkado todos os arquivos .js (scripts jQuery e javascript) e .css (formatação e estilo dos componentes do template) que serão utilizados na aplicação, além da construção do modelo das páginas, como rodapé, menu principal e cabeçalho do sistema, deixando para os arquivos de interface somente a parte específica de cada um.  O mesmo será feito no back-end do projeto App, adicionando qualquer função que será comum à todos os arquivos .cs no arquivo .cs da Master Page.  Quanto à organização, ambos os projetos serão modulados, ou seja, as funcionalidades do sistema serão agrupadas em pastas de acordo com sua função, cada grupo de funções também terá um namespace diferente, ou seja, para chama-las, será necessário ao desenvolvedor especificar a qual módulo ela pertence.  O projeto Lib será referenciado pelo projeto App, isso significa criar uma ponte, um link de acesso, para que a interface “converse” com a lógica do sistema.  **Desenvolvimento do Web Site (Projeto Lib)**  De acordo com o padrão MVC, o projeto Lib será responsável pela lógica do sistema e comunicação deste com o banco de dados, funcionando como uma camada entre o usuário e o servidor de dados.  A organização do projeto, além de modularizada, também contará com um arquivo principal que contém propriedades e funções comum a todos, como, por exemplo, conexão com banco de dados, dados de acesso de usuário e funções de log do sistema.  Este projeto possuíra referência ao .Net Framework e bibliotecas de sistema, comunicação com o servidor, linguagem e de controle de transações. Essas bibliotecas são necessárias ao sistema, no sentido de desenvolvimento e controle entre interações com os dados do servidor.  **Comunicação externa**  Após criado o projeto, comunicação entre interface e servidor estabelecidas, será dado início ao desenvolvimento da comunicação entre os microcontroladores arduíno. Para esta etapa será necessário configurar tanto o projeto App, quanto os microcontroladores.  No projeto, toda comunicação externa será realizada a partir de funções e eventos em jQuery nas próprias páginas do website ou por arquivos .asmx, que utilizam o padrão WebService do ASP.Net, este padrão funciona como uma ponte entre aplicações, no caso a aplicação do projeto e as aplicações de cada microcontrolador. Os WebServices utilizam arquivos XML para fazer a comunicação, ou seja, ele recebe e envia um pacote XML contendo as informações necessárias, o destinatário deste pacote irá processar as informações e realizar as funções requisitadas. O XML é um padrão de arquivo fácil de manipular, por esse motivo foi escolhida esta tecnologia.  **Configuração no microcontrolador:**  Os microcontroladores deverão estar configurados para estabelecer a comunicação com o projeto, sendo assim possível receber e enviar pacotes de informações via rede local.  **Configuração no projeto:**  (POST) Uma requisição do sistema funcionará da seguinte forma: A página web irá requisitar alguma informação de um microcontrolador específico através de um função em jQuery, a função terá os parâmetros necessários e o endereço IP do microcontrolador de destino, está função irá enviar um pacote de informações formatadas para o endereço do microcontrolador.  O microcontrolador será responsável pelo processamento das informações e entrega do retorno para o sistema.  O sistema informará ao usuário se a requisição foi ou não realizada com sucesso.  (GET) O recebimento de pacotes pelo sistema será enviado pelo microcontrolador ao sistema através de uma url direta ao webservice, passando os parâmetros e valores necessários. O sistema irá processar essas informações e enviar uma resposta ao microcontrolador.  Por todos os componentes de hardware estarem ligados em uma rede local, faz-se necessário desabilitar o firewall, para que o mesmo não bloqueie o envio e recebimento de pacotes na rede, futuramente, está previsto que o sistema seja acessível via internet, nesse caso, o projeto será adaptado para atender os requisitos de comunicação e segurança.  **Modelagem PIM**  **Casos de Uso**  **Diagrama de Requisitos**  **Diagrama de Bloco**  **Bloco de Servidor Interno** |

|  |
| --- |
| **REFERÊNCIAS BIBLIOGRAFICAS**  [https://msdn.microsoft.com/en-us/library/bb165951%28v=vs.140%29.aspx](https://msdn.microsoft.com/en-us/library/bb165951(v=vs.140).aspx)  <https://msdn.microsoft.com/en-us/library/ff649643.aspx>  [http://en.wikipedia.org/wiki/Bootstrap\_%28front-end\_framework%29](http://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))  [http://en.wikipedia.org/wiki/Bootstrap\_%28front-end\_framework%29](http://en.wikipedia.org/wiki/Bootstrap_(front-end_framework))  <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6698009&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D6698009>  <https://msdn.microsoft.com/pt-br/library/cc580600.aspx>  [https://msdn.microsoft.com/en-us/library/t745kdsh%28v=vs.90%29.aspx](https://msdn.microsoft.com/en-us/library/t745kdsh(v=vs.90).aspx)  Modelagem PIM  • Três modelos PIM (Platform Independent of Models)  o Casos de Uso  o Modelo Hierárquico de requisitos  o Modelo comportamental (com anotações MARTE)  • Iniciar o Desenvolvimento do protótipo  Entrega (artigo) e Apresentação (protótipo) do Projeto final – 03/06/2015  • Modelos PSM (Platform Specific od Model)  o Modelo de Bloco de Definição (com anotações MARTE)  o Modelo de Bloco Interno (com anotações MARTE)  Embasamento Teórico, Artigo (descrever resumidamente conceitos, técnicas, ferramentas, ... abordadas no projeto – usar ao menos duas referências para cada assunto).  Os itens abaixo dedem ser desenvolvidos com um artigo conforme o modelo em:  goo.gl/Iz8lZf  Capitulo deverá conter:  Título  Seções  Subseções (se necessário)  Considerações finais  Resultados obtidos.  Referências bibiográficas.  Páginas com anexos (se necessário). |