

Engenharia da Computação - ESS/EC Cin UFPE

Plano de Projeto - Auto House

Introdução

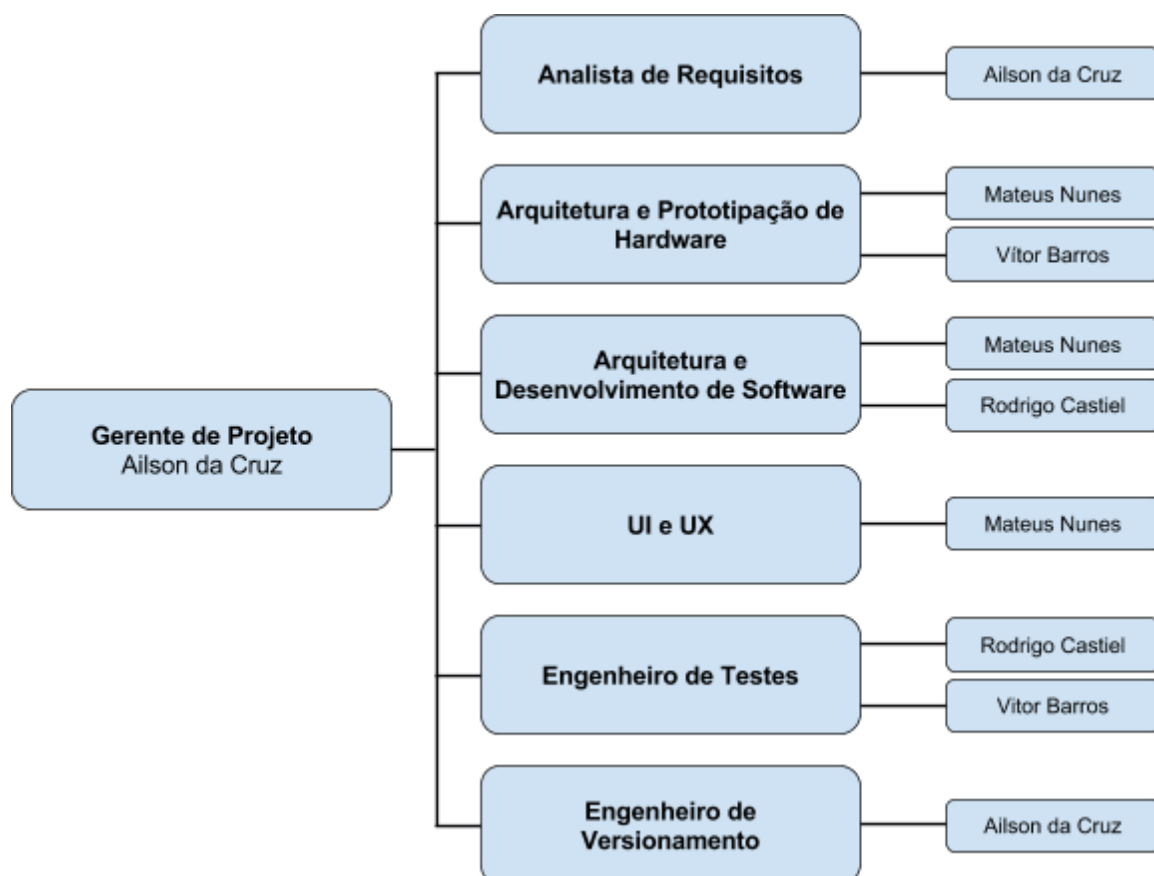
O projeto Auto House visa o desenvolvimento de soluções tecnológicas capazes de automatizar hábitos executados rotineiramente em ambientes residenciais. Este plano de projeto, em especial, trata do desenvolvimento de um controlador de persianas com uma interface de controle remoto através de redes Wi-Fi.

Visando melhorar a qualidade de vida das pessoas, o produto poderá ser programado para funcionar de forma autônoma e em acordo com a conveniência dos usuários. Eliminando quase que totalmente a operação manual de persianas e reduzindo consideravelmente o tempo gasto para controlá-las.

Organização do Projeto

Desenvolvedores:

- Ailson da Cruz (acgt@cin.ufpe.br)
- Mateus Nunes (mnbm@cin.ufpe.br)
- Rodrigo Castiel (rcrs2@cin.ufpe.br)
- Vítor Barros (vba2@cin.ufpe.br)



Área	Atribuições
Gerente de Projeto	Definir, coordenar e integrar as atividades executadas para o desenvolvimento do projeto.
Analista de Requisitos	Determinar, definir e gerenciar o estado e demais aspectos relacionados aos requisitos de software do sistema.
Prototipação de Hardware	Projetar a arquitetura e desenvolver o hardware do projeto realizando os testes pertinentes.
Desenvolvimento de Software	Projetar a arquitetura e desenvolver os componentes de software utilizados no projeto, realizando testes sob demanda.
UI e UX	Concepção das interfaces e sua integração com os componentes de software além da realização de testes sob demanda.
Engenheiro de Testes	Criação de estratégias de teste com objetivo de validar os componentes do sistema durante e após seu desenvolvimento.
Engenheiro de Versionamento	Controle e concatenação de versões dos módulos do projeto.

Requisitos do Projeto e de Recursos

Requisitos do projeto:

- O sistema deve ser capaz de abrir e fechar as lâminas da persiana;
- O sistema deve ser capaz de ligar e desligar a lâmpada;
- O sistema deve ser capaz de controlar a luminosidade, caso, o modelo da persiana dê suporte a esta funcionalidade;
- O sistema deve ser capaz de agir sozinho, a partir das informações dos sensores e configurações do usuário, controlando a luminosidade;
- O sistema deve ser capaz de agir sozinho, a partir das informações do despertador do usuário;
- O sistema deve ser compacto e leve;
- O sistema deve ter um baixo consumo de energia;
- O sistema deverá oferecer uma interface de controle através de conexão Wi-Fi;
- O usuário deve ser capaz de controlar remotamente as funcionalidades do sistema;
- O usuário deve ter o status atual do sistema;

- O usuário deve poder desligar/ligar o sistema;
- O usuário deve receber notificações quando houver;
- Para funcionar, o sistema deverá ser configurado por um dispositivo com conexão Wi-Fi. Na configuração inicial, será criada uma senha de segurança para operar o sistema;
- O sistema deverá possuir memória para armazenar configurações de operação pré-definidas pelo usuário;

Requisitos de recursos:

- Raspberry PI 3: será utilizado para integrar e simular o hardware necessário para o projeto;
- Protoboard: Placa de ensaio para montagem do circuito do projeto, integrando os componentes de hardware para o funcionamento em conjunto;
- Resistores e jumpers: Necessário para a montagem do circuito na protoboard;
- Sensor de luminosidade: Necessário para correta funcionalidade da automação da cortina, sabendo a quantidade de luz do ambiente.
- Motor de passo: Motor para controlar a abertura e fechamento da cortina;
- Módulo relé: Necessário para a automação da luz;
- Cortina horizontal ou persiana: para teste do sistema final;
- Servidor: para comunicação do sistema com o aplicativo de controle.
- Macbook: sistema computacional para desenvolvimento de aplicação iOS;
- iPhone: dispositivo móvel para simulação do sistema em desenvolvimento;
- Xcode: ambiente padrão de desenvolvimento em Swift/Objective-C para software controlador do usuário.

Análise de riscos

Essa análise tem como objetivo identificar os riscos do projeto.

Cada risco identificado é atribuído um valor de impacto (efeito que o risco pode causar ao projeto) e um valor de probabilidade (chance de ocorrer o erro). Esses atributos podem assumir valores em uma escala definida de 1 a 3, sendo o cálculo do risco dado pela multiplicação desses valores.

Depois do levantamento dos riscos e utilizando os valores mencionados acima, foi formada a seguinte tabela:

Código	Descrição	Probabilidade	Impacto	Risco
R1	Integração de software e hardware.	3	3	9
R2	Integração de componentes de hardware com a estrutura física.	3	2	6
R3	Inconsistência de comunicação entre módulos	3	2	5
R4	Aprendizagem de novas tecnologias.	3	2	6

R5	Mudanças de requisitos.	2	2	4
R6	Abandono ou falta de produtividade de integrante.	1	3	3

A seguir tem as descrições mais detalhada dos riscos, assim como ações para mitigar a ocorrência do risco e o plano de ação caso o risco ocorra.

R1 - Integração de software e hardware

Problemas para integrar os componentes de hardware e software do projeto.

- **Ações para mitigar o risco:** Escolher tecnologias já conhecidas pelos desenvolvedores ou que já foram usadas em projetos semelhantes, verificando as possibilidades de compatibilidades entre as tecnologias.
- **Plano de ação:** Atribuir mais desenvolvedores para tentar sanar os problemas de integração. Caso não seja possível, procurar tecnologias semelhantes para substituição ou mudar na arquitetura de hardware/software do projeto para se adequar a tecnologias conhecidas e disponíveis.

R2 - Integração de componentes de hardware com a estrutura física

Problemas para integrar os hardwares de automação do projeto as cortinas e lâmpadas do ambiente.

- **Ações para mitigar o risco:** Projetar esquemáticos de conexões mecânicas e peças, assim como fazer o estudo para melhor escolher os componentes e tipos de cortinas suportadas.
- **Plano de ação:** Procurar por tecnologias alternativas para ser aplicado, como diferentes motores e diferentes tipos de cortinas.

R3 - Inconsistência de comunicação entre módulos

Máquinas de estado incoerentes entre dispositivos móveis e sistemas computacionais de cortina e lâmpada.

- **Ações para mitigar o risco:** Projetar esquemas de comunicação e simular protocolos na camada de aplicação para encontrar possíveis pontos de falha;
- **Plano de ação:** Estudar melhores soluções (bibliotecas e APIs) de comunicação ou cuidadosamente corrigir a arquitetura e protocolos.

R4 - Aprendizagem de novas tecnologias

Atrasos devido a aprendizagem de novas tecnologias para realizar o projeto.

- **Ações para mitigar o risco:** Atribuir pessoas com conhecimento prévio na tecnologia ou em tecnologias semelhantes. Assim como a escolha eficiente de quais tecnologias são realmente necessárias para o projeto e que são de mais fácil aprendizagem e desenvolvimento.
- **Plano de ação:** Verificar se pode ser substituída por outra tecnologia semelhante de mais fácil desenvolvimento. Caso seja uma tecnologia crítica para o projeto, alocar mais desenvolvedores para tentar superar a dificuldade e mitigar o atraso.

R5 - Mudanças de requisitos

Mudanças nos requisitos do projeto e de requisitos de recursos de hardware e software.

- **Ações para mitigar o risco:** Planejar e definir requisitos em acordo com o cliente (com o usuário final), fazendo estudo de viabilidade das funcionalidades.
- **Plano de ação:** Tentar minimizar a mudança para diminuir o impacto e verificar o impacto nos demais requisitos, procurando minimizar as chances de novas mudanças.

R6 - Abandono ou falta de produtividade de integrante

Saída de integrante por desistência da participação no projeto ou falta de produtividade por desinteresse no projeto.

- **Ações para mitigar o risco:** Estimular membros e sempre dividir as atividades de acordo com os perfis correspondentes.
- **Plano de ação:** Caso haja desistência, procura-se alocar as tarefas a outros desenvolvedores, assim como a colaboração do integrante para ajudar na transição da tarefa para diminuir as chances de problema por falta de conhecimento. Caso a produtividade caia, descobrir razão e contornar com realocação ou divisão de trabalho.

Estrutura analítica

A fim de facilitar o processo de gerenciamento do projeto e agilizar o desenvolvimento do mesmo, foi elaborada uma estrutura analítica com base nas discussões realizadas pelo grupo a priori de estudos mais detalhados.

W0 - Definição do Plano de Projeto

Planejamento do projeto, definindo atribuições dos desenvolvedores, requisitos, tarefas e riscos. Assim com os marcos de entregas para o acompanhamento do projeto.

W1 - Definição dos recursos de hardware e software que vão ser utilizados para o desenvolvimento do projeto.

Faz-se necessário o estudo (pesquisa de preço e disponibilidade no mercado), a definição e a obtenção de recursos (sensores, motores, frameworks, linguagem de programação) que vão ser utilizados no desenvolvimento e concepção do projeto.

W2 - Modelagem arquitetural do sistema e da comunicação entre os componentes de internos.

Após a aquisição dos componentes necessários e definição das tecnologias a serem utilizadas, deverá ser definida a arquitetura do sistema e a interface de comunicação entre os componentes do mesmo.

W3 - Implementação da interface de integração dos componentes físicos com os modelos de persianas suportados.

Criação de protótipos físicos do produto para facilitar a realização de testes do software sendo implementado para operar as persianas. Implementação das funcionalidades básicas com participação direta dos componentes de hardware, como a abertura e fechamento das persianas.

W4 - Integração dos módulos de sensores e atuadores.

Integrar os sensores de luminosidade e o módulo relé ao circuito projeto.

W5 - Implementação do aplicativo para dispositivos móveis, pelos quais ocorrerá o controle do produto.

Definir as interfaces de controle necessárias para que o usuário final tenha facilidade no uso das funcionalidades oferecidas pelo produto. Esta tarefa deve ser bem especificada para que nenhuma funcionalidade seja ignorada ou receba menos atenção, provocando insatisfação nos consumidores.

W6 - Implementação da camada de comunicação entre o hardware presente no controlador de persianas e o aplicativo para dispositivos móveis através de redes Wi-Fi e Bluetooth.

Esta tarefa tem como finalidade a implementação dos mecanismos necessários para garantir a comunicação confiável e segura entre os componentes físicos que controlam as persianas e a aplicação móvel. Esta camada de comunicação também servirá para a troca de informações provenientes de sensores presentes no hardware da persiana.

W7 - Realização de testes, correção de bugs e eventuais melhorias.

Contempla a realização de testes exaustivos a fim de descobrir eventuais falhas, identificar dificuldades e propor melhorias na utilização do projeto como um todo. Para execução desta atividade, é fundamental a realização de todas as tarefas supracitadas.

W8 - Entrega do primeiro release estável do projeto.

Após a realização de testes exaustivos e eliminação de eventuais bugs, a primeira versão estável do projeto estará pronta para publicação. Ela deverá funcionar de acordo com a visão de produto proposta e executar satisfatoriamente bem as funcionalidades prometidas.

Cronograma do projeto

A tabela abaixo, estipula os prazos pretendidos para a realização das atividades propostas na estrutura analítica do projeto.

Atividade	Dependências	Prazo	Responsáveis
W0	-	05/10/2016	acgt, mnbm, rcrs2, vba2
W1	W0	26/10/2016	mnbm, rcrs2, vba2
W2	W1	26/10/2016	mnbm, rcrs2, vba2
W3	W1 e W2	26/10/2016	mnbm, vba2
W4	W1 e W2	21/10/2016	vba2
W5	W2	21/11/2016	mnbm, rcrs2
W6	W3 e W5	21/11/2016	mnbm, rcrs2
W7	W4 e W6	05/12/2016	rcrs2, vba2
W8	W7	05/12/2016	acgt, mnbm, rcrs2, vba2

Mecanismos de monitoramento e elaboração de relatórios

O desenvolvimento do projeto vai acontecer a partir de sprints com uma duração média de 2 semanas. Ao fim de cada sprint um conjunto de entregáveis será utilizado para o acompanhamento do projeto.

Com base nas atividades definidas neste relatório, assim como o cronograma, serão traçados os marcos e entregáveis para o desenvolvimento incremental do projeto, definindo prioridade, dependência, prazos e responsáveis pelo desenvolvimento. Assim diminuindo as possíveis dificuldades e mitigando os possíveis riscos do projeto.

Para auxiliar o acompanhamento do projeto, serão utilizadas algumas ferramentas, como o Trello e o Github. O primeiro proporciona o acompanhamento organizado das tarefas e o segundo proporciona uma gestão de código e de versionamento.