

Laboratório de Projeto - Concepção, Gestão e Implementação de Projetos Aplicados ao Ensino de Engenharia de Sistemas^{*}

Renata Nésio^{*} Pedro F. Freitas^{*} Gustavo M. Freitas^{*}

^{*} *Escola de Engenharia - Universidade Federal de Minas Gerais*
e-mails: (renatanesio,fernandesp,gustavomfreitas)@ufmg.br

Abstract: This article presents the teaching methodology applied in the discipline Project Laboratory III of the Systems Engineering course at the Federal University of Minas Gerais. During the discipline, students develop a non-trivial project, involving software and hardware integration, based on a case study, from the conception to the implementation of a functional prototype. To this end, students must apply Requirements Engineering concepts, and use concept maps and finite state machines to obtain a holistic view of the project, whose development is managed through Scrum and Kanban agile management methodologies. The case study presented consists on constructing an automatic crane, integrating contents from mechanics and 3D modeling, electronics and software. During the execution of the project, the students use free software and hardware to implement a low-cost prototype, enabling the use of the proposed didactic methodology in other public universities and institutions in the country.

Resumo: Este artigo apresenta a metodologia de ensino aplicada na disciplina Laboratório de Projeto III do curso de graduação de Engenharia de Sistemas da Universidade Federal de Minas Gerais. Durante a disciplina, os alunos desenvolvem um projeto não trivial, envolvendo a integração de software e hardware, com base em um caso de estudo, desde sua concepção até a implementação de um protótipo funcional. Para tal, os alunos devem aplicar conceitos de Engenharia de Requisitos, e empregar mapas conceituais e máquinas de estado finito para obter uma visão holística do projeto, cujo desenvolvimento é gerenciado através das metodologias Scrum e Kanban de gestão ágil. O caso de estudo apresentado consiste na construção de um guindaste automático, integrando conteúdos de mecânica e modelagem 3D, eletroeletrônica e software. Durante a execução do projeto, são utilizados software gratuito e hardware livre para a implementação de um protótipo em escala de baixo custo, viabilizando o emprego da metodologia didática proposta em outras universidades e instituições de ensino públicas do País.

Keywords: Teaching Engineering; Project-Based Learning; Systems Integration; Automated Crane.

Palavras-chave: Educação em Engenharia; Aprendizagem Baseada em Projeto; Integração de Sistemas; Guindaste Automatizado.

1. INTRODUÇÃO

O curso de Engenharia de Sistemas (ES) na Universidade Federal de Minas Gerais (UFMG) foi criado em 2010 e possui disciplinas fundamentais para a formação de um Engenheiro de Sistemas. Diversas definições para a Engenharia de Sistemas compartilham os conceitos da abordagem de sistemas, como a visão holística, a síntese e a importância das inter-relações, assim como ideias baseadas em projeto de engenharia do ciclo de vida e dos requisitos do sistema (Ramos et al., 2012).

A complexidade estrutural do conhecimento científico-tecnológico e o crescente impacto da engenharia em decisões na sociedade e no meio ambiente demandam o rápido crescimento do ensino de engenharia. Alguns problemas

críticos na educação em engenharia identificados pela indústria são listados por Mills and Treagust (2003), como currículos focados nos cursos de ciências e técnicas de engenharia sem fornecer uma integração suficiente desses tópicos ou relacioná-los à prática industrial, assim como a falta de habilidades de comunicação e experiências em trabalho em equipe.

Neste contexto, a Aprendizagem Baseada em Projeto (ABP) surge como uma abordagem de ensino que estimula a solução de problemas práticos. A ABP é orientada pelo aluno e facilitada pelo professor; nela, cada passo é supervisionado e aprovado e um projeto é criado a partir da investigação (Bell, 2010). Um exemplo de aplicação da ABP é a proposta de Robótica Educacional apresentada em (Denis and Hubert, 2001), onde os próprios alunos são responsáveis por construir e programar robôs em escala.

Buscando uma metodologia de ensino que concilie teoria e prática de forma integrada, a disciplina Laboratório de

^{*} Este trabalho teve apoio do Colegiado de Engenharia de Sistemas e Escola de Engenharia da Universidade Federal de Minas Gerais (UFMG), e do Instituto Tecnológico Vale (ITV).

Projeto III emprega conceitos da Aprendizagem Baseada em Projetos, e utiliza como referência as ideias construtivistas de Jean Piaget sobre a participação ativa do aluno no processo de aprendizado (Piaget et al., 2006).

Um dos objetivos da disciplina é preencher lacunas no ensino através do desenvolvimento de um projeto proposto como estudo de caso. Nela, os alunos participam de todas as etapas de um projeto não trivial, partindo da concepção da ideia até a entrega de um protótipo funcional. A disciplina Laboratório de Projeto III também contribui na formação do conhecimento técnico, capacitando o aluno nos processos de documentação, gestão e implementação de um projeto não trivial. Além disso, são colocados em prática conceitos teóricos abordados ao longo do curso de Engenharia de Sistemas, como levantamento de requisitos, gestão ágil, construção de projeto eletroeletrônico e desenvolvimento de software. Um requisito fundamental da disciplina é a utilização de hardware livre e software gratuito para a implementação de um protótipo em escala de baixo custo.

O restante deste artigo está organizado na seguinte forma: na Seção 2 são apresentadas a proposta da disciplina Laboratório de Projeto III e as ferramentas utilizadas para o desenvolvimento do projeto; na Seção 3 estão descritos os conteúdos abordados na disciplina; os resultados obtidos pelos alunos são apresentados na Seção 4; por fim, as conclusões e trabalhos futuros são apontados na Seção 5.

2. PROPOSTA DA DISCIPLINA

O curso de Engenharia de Sistemas da UFMG possui cinco disciplinas de laboratório fundamentais para o desenvolvimento do pensamento sistêmico em seus alunos. Os Laboratórios de Projeto, cursados entre o 5º e o 9º período, orientam alunos para desenvolver projetos que integram conteúdos vistos ao longo do curso e trabalham habilidades relacionadas à especificação, projeto, verificação e validação, e análise de sistemas (Santos et al., 2013).

A disciplina de Laboratório de Projeto III, prevista na grade curricular para o 7º período, propõe a concepção, gestão e implementação de um protótipo funcional, atendendo a uma série de requisitos especificados no início do semestre que devem ser cumpridos até a data da apresentação. Em cada semestre, um caso de estudo diferente é utilizado como tema do projeto.

Um diferencial da metodologia de ensino proposta é o conjunto de ferramentas utilizadas pelos alunos para a análise e modelagem de sistemas, assim como gerenciamento de projetos. Primeiramente, os alunos devem realizar a Engenharia de Requisitos e organizar as principais ideias em mapas conceituais. Com os requisitos levantados, os alunos constroem uma Máquina de Estados Finitos (FSM) a fim de verificar os estados e as transições do sistema antes de sua implementação. Todo o projeto é gerenciado com uma combinação das metodologias ágeis Scrum e Kanban.

As atividades são realizadas em grupos de três ou quatro alunos, formados pelo professor de forma que cada grupo possua membros com habilidades diversificadas e seja capaz de cumprir todas as etapas do projeto. Ao final de cada etapa, os grupos apresentam documentações descritivas das atividades realizadas ao professor, que aponta as corre-

ções necessárias. Ao final do semestre, cada grupo entrega um relatório técnico com toda a documentação do projeto, e realiza uma demonstração do protótipo implementado.

Cabe observar que, mesmo o projeto não englobando todo o escopo da Engenharia de Sistemas, o trabalho proposto tem como foco o ensino de uma abordagem sistêmica, e insere os alunos em um contexto de desenvolvimento e aplicação da ES.

2.1 Engenharia de Requisitos

Segundo Faisandier (2013), requisito é uma declaração que identifica uma característica esperada de um produto, serviço, empresa ou processo operacional, funcional ou de restrição, inequívoca, testável ou mensurável e necessária para a aceitação do produto, serviço, empresa ou processo. Sommerville (2005) descreve algumas atividades fundamentais para os processos de Engenharia de Requisitos, sendo elas: elicitação, análise, validação, negociação, documentação e gestão.

Como todo sistema não trivial, o projeto da disciplina Laboratório de Projeto III possui uma série de requisitos que devem ser atendidos ao final do semestre. Desta forma, a primeira atividade realizada pelos alunos consiste em levantar os requisitos funcionais e não funcionais necessários para a implementação do sistema.

2.2 Mapas Conceituais

Mapas conceituais são ferramentas gráficas que representam visualmente as relações entre conceitos e ideias. Estes têm sido utilizados de forma eficaz no auxílio da aprendizagem significativa, resultando na modificação de estruturas de conhecimento do aluno (Novak, 2002).

O mapa conceitual facilita compreender os requisitos, sintetizar e integrar conceitos, assim como promove o aprendizado colaborativo do time. O aspecto visual também promove a compreensão holística do sistema e simplifica a etapa de verificação.

Para respeitar o princípio de utilizar software gratuito, recomenda-se aos alunos o uso do CmapTools. Durante a disciplina, os alunos apresentam três mapas conceituais: geral, hardware e software. Esses promovem uma compreensão geral do projeto antes de sua implementação.

2.3 Máquinas de Estado

Existem várias formas de representar o funcionamento de um sistema; entre elas, a Máquina de Estados Finitos (FSM) é uma das mais conhecidas. Uma FSM consiste em estados que definem um comportamento, transições de um estado para outro, e eventos que podem gerar a transição entre estados (Jezernik et al., 2012). Diagramas FSM ilustram as transições, os eventos que servem como gatilho, as condições necessárias para uma transição e os possíveis estados de um sistema.

Na disciplina de Laboratório de Projetos III, uma das etapas para a concepção do projeto é a criação de um diagrama FSM, que permite visualizar todos os estados do sistema e possibilita analisar, verificar e gerenciar todos os requisitos estabelecidos para o projeto proposto.

2.4 Gestão Ágil de Projetos

Para o gerenciamento do projeto, os alunos fazem uso das metodologias ágeis Scrum e Kanban. Essas metodologias têm como objetivo flexibilizar o processo de desenvolvimento de software, permitindo assim que as equipes tenham um desempenho mais eficaz (Maruping et al., 2009). Embora essa abordagem tenha sido criada para equipes de software, ela se aplica a diversos cenários. A gestão ágil é utilizada na disciplina para auxiliar no gerenciamento de todas as etapas do projeto.

Scrum é um framework de metodologia ágil que segue princípios e valores estabelecidos no Manifesto Ágil. As equipes ágeis não seguem um plano de ação rigidamente definido ao longo de um projeto; em vez disso, elas trabalham em ciclos, usando iterações frequentes e pontuais que permitem check-ins regulares com feedback de seus colegas e clientes do produto final (Krehbiel et al., 2017). O ciclo de desenvolvimento do Scrum é apresentado na Figura 1. Em conjunto, aplica-se a prática do Kanban, um subsistema do Sistema Toyota de Produção (TPS). Um dos principais artefatos dessa metodologia é o Kanban Board, que se divide em três colunas: atividades a fazer, atividades em desenvolvimento e atividades concluídas. Agregar as duas metodologias ajuda os alunos a ter uma visão holística do projeto e cumprir tarefas de forma ágil.

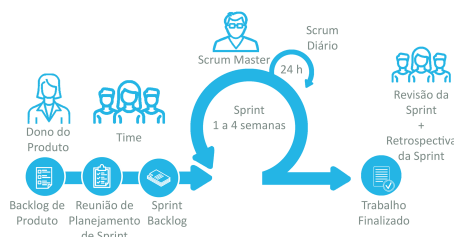


Figura 1. Ciclo de desenvolvimento do Scrum.

Na disciplina de Laboratório de Projeto III, os alunos adaptam o modelo do Scrum para reuniões semanais, e registram todas as atividades desenvolvidas durante o projeto na plataforma gratuita Trello¹ aplicando a metodologia Kanban.

2.5 Caso de Estudo: Guindaste Automatizado

A cada semestre é apresentado um caso de estudo diferente para os alunos da disciplina Laboratório de Projeto III da UFMG. Este artigo apresenta o caso proposto para turma do segundo semestre de 2019: a construção de um guindaste automatizado.

Guindastes são máquinas criadas para içar, mover e abaixar objetos pesados através de um braço móvel. Os guindastes podem ser classificados com base nos graus de liberdade que o mecanismo de suporte oferece no ponto de suspensão (Abdel-Rahman et al., 2003).

Durante a disciplina, os alunos devem projetar e implementar um guindaste tipo torre em escala. O guindaste automatizado deve ser capaz de içar e movimentar uma moeda de cinquenta centavos com um eletroímã, ter alcance e altura entre 25 e 35 cm, e girar de -180° a $+180^\circ$.

¹ <https://trello.com/pt-BR>

Como restrições de projeto, foi imposto que os alunos utilizassem palitos de picolé para construir a estrutura do guindaste, um microcontrolador Arduino para acionar o sistema eletrônico, e softwares gratuitos para o gerenciamento do projeto e desenvolvimento da interface gráfica de usuário.

A construção de um protótipo funcional em escala utilizando materiais acessíveis e hardware e software livres reduz os custos associados à disciplina, viabilizando o emprego das práticas didáticas propostas em outras universidades e instituições de ensino públicas do País.

3. CONTEÚDOS ABORDADOS

Nesta seção estão descritos os principais conteúdos abordados pelos alunos da disciplina Laboratório de Projeto III durante o desenvolvimento e implementação do protótipo, que envolve a construção da estrutura mecânica e do circuito elétrico acionado por microcontrolador, e o desenvolvimento de uma Interface Gráfica de Usuário (GUI) mais software embarcado.

3.1 Projeto Mecânico e Construção

Esta etapa diz respeito ao projeto e construção da estrutura física do protótipo.

A primeira etapa consiste em modelar a estrutura do guindaste utilizando uma aplicação CAD. Para cumprir a exigência de utilizar aplicações e plataformas gratuitas, é indicado o uso da versão gratuita do SketchUp², um software próprio para a criação de modelos tridimensionais.

A implementação da estrutura física deve ser orientada pelo modelo 3D desenvolvido. Segundo os requisitos, ela deve ser composta por palitos de picolé. Nesta etapa de construção é importante planejar quais partes podem ser construídas separadamente e como estas partes serão unidas posteriormente. Após construir cada parte da estrutura, é necessário verificar se elas apresentam a rigidez desejada; a mesma verificação é repetida após a integração de todas as partes do guindaste.

3.2 Projeto Eletroeletrônico e Acionamentos

Um dos objetivos da disciplina Laboratório de Projeto III também é aplicar conhecimentos adquiridos em disciplinas de circuitos elétricos e eletrônica. Para rotacionar a lança do guindaste e içar o eletroímã, os alunos precisam especificar o tipo de motor a ser utilizado: motor servo, motor de passo ou motor de corrente contínua com encoder. É necessário que os alunos estudem as características de cada um e optem pelo que melhor se aplica ao projeto. O eletroímã também deve ser escolhido pelos alunos, atendendo à exigência de carga mínima.

Os motores e eletroímã são acionados via Arduino, uma placa criada para ser robusta, de baixo custo, código facilmente configurável e hardware livre (Nussey, 2018). Todo o código de acionamento também é desenvolvido pelos alunos.

² <https://www.sketchup.com/pt-BR>

Nesta etapa do projeto, é importante avaliar as características dos motores e do eletroímã escolhidos para a correta especificação das fontes de alimentação, os circuitos de acionamento e a integração entre eles e o Arduino.

No caso dos motores, além de uma fonte de alimentação com tensão compatível e capaz de fornecer a corrente demandada, pode ser necessária a especificação de um circuito de interface e potência para realizar o acionamento desses atuadores.

Para o acionamento do eletroímã é necessário utilizar uma fonte de alimentação compatível com seus níveis de tensão e corrente de operação, e especificar um circuito capaz de ativá-lo e desativá-lo. Esse circuito deve apresentar mecanismos de regulação de corrente e proteção para corrente de retorno, evitando assim possíveis danos ao Arduino e demais componentes.

3.3 Interface Gráfica de Usuário, Software Embarcado e Comandos

Na disciplina Laboratório de Projeto III, também é possível colocar em prática a teoria de algoritmos vista nas disciplinas de Algoritmos e Estrutura de Dados, assim como em Programação Orientada a Objeto, e desenvolver aplicações *web*. Nesta etapa do projeto, os alunos devem desenvolver uma interface gráfica para que o usuário do guindaste seja capaz de operá-lo sem necessariamente compreender a engenharia por detrás do projeto.

A interface deve ser amigável, executável em um computador ou celular e capaz de enviar comandos para rotacionar a lança em ângulos específicos, alterar a altura e ligar ou desligar o eletroímã. Além disso, o sistema dessa interface deve ser composto por *front-end* - parte gráfica onde os *inputs* do usuário são recebidos e informações do guindaste são exibidas, e *back-end* - responsável por realizar a troca de mensagens com o Arduino.

Para realizar o controle do guindaste, um software embarcado no Arduino deve ser desenvolvido. Este deve ser capaz de: receber mensagens oriundas da interface gráfica e interpretá-las, identificando qual a ação a ser executada; controlar o movimento dos motores e acionar o eletroímã para que as ações pretendidas sejam corretamente executadas; e, por fim, enviar mensagens para a interface gráfica, sinalizando o término das ações e indicando se elas foram bem sucedidas ou não.

Uma exigência aos alunos foi a criação de um repositório para o projeto no GitHub onde todos os códigos desenvolvidos devem ser adicionados, facilitando assim o controle de versões e permitindo o desenvolvimento colaborativo dos softwares.

3.4 Protocolo de Comunicação

Protocolos de comunicação são descrições formais de formatos e regras de mensagens digitais; eles são obrigados a trocar mensagens dentro ou entre sistemas de computadores. Um dos requisitos de projeto imposto na disciplina foi o desenvolvimento de um protocolo utilizando como base mensagens de até dois bytes.

No projeto guindaste automatizado, o protocolo de comunicação serve como um meio de transmissão de dados entre o *back-end* da interface gráfica e o Arduino.

Nesta etapa, é necessário definir uma codificação que utilize dois bytes e que permita identificar a ação a ser executada, sendo elas rotação, mudança de altura ou acionamento do eletroímã, e qual o estado pretendido, que por sua vez pode ser o ângulo de rotação, altura ou estado do eletroímã.

É importante lembrar que essa comunicação deve ocorrer nos dois sentidos: da interface gráfica para o controlador, e do controlador para a interface gráfica. Assim, a codificação também é utilizada pelo controlador para informar ao usuário do fim de uma ação, o resultado desta e o estado do guindaste.

4. RESULTADOS OBTIDOS PELOS ALUNOS

Ao final da disciplina, além de realizar uma demonstração do protótipo, cada grupo de alunos deve entregar um relatório técnico descrevendo a concepção, desenvolvimento e gestão de todo o projeto. A seguir são apresentados os resultados obtidos pelo “Grupo E” de alunos do segundo semestre de 2019, que teve como integrantes os co-autores deste artigo Renata Nésio e Pedro Freitas.

4.1 Requisitos Operacionais e Mapas Conceituais

Uma das primeiras tarefas realizadas pelos alunos é a Engenharia de Requisitos. Eles determinam os requisitos funcionais e não funcionais para o projeto de guindaste automatizado, aplicando conceitos aprendidos nas disciplinas de Laboratório de Projeto I e Engenharia de Software.

A seguir, são listados como exemplo alguns dos requisitos de software levantados pelo Grupo E:

- o guindaste automatizado deve possuir um módulo de interface que se comunica via protocolo com um algoritmo embarcado em Arduino;
- o protocolo de comunicação deve se limitar a dois bytes;
- o módulo de interface deve possuir seletores de ângulo de rotação e altura do eletroímã e um botão de acionamento do eletroímã;
- o algoritmo embarcado deve possuir módulos de acionamento do eletroímã, acionamento dos motores e leitura do estado dos motores.

Nesta etapa também são criados mapas conceituais que auxiliam no levantamento de requisitos. O mapa conceitual geral do guindaste automatizado é apresentado na Figura 2. A partir dos mapas, os alunos são capazes de refinar os requisitos funcionais e não funcionais do sistema.

4.2 Máquina de Estados Finitos do Sistema

A fim de verificar o projeto de funcionamento do guindaste automatizado e as transições entre seus estados, os alunos elaboram um diagrama de máquina de estados finitos, como apresentado na Figura 3. De maneira geral, o guindaste pode estar em um dos seis estados possíveis, e as transições ocorrem a cada requisição enviada para o controlador.

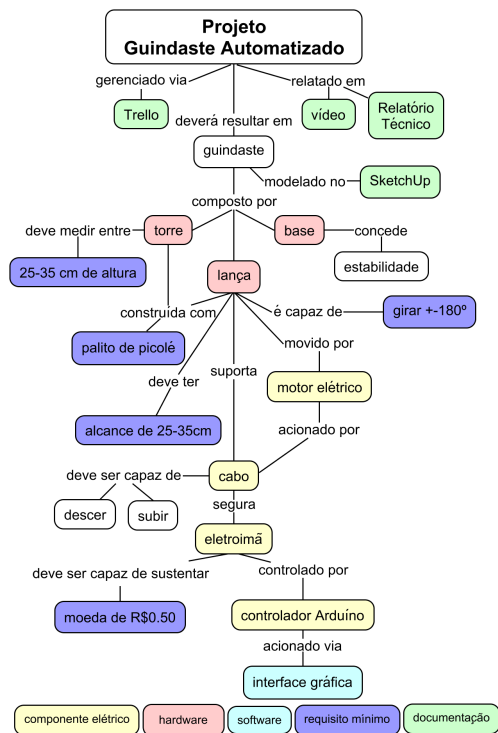


Figura 2. Mapa conceitual geral do guindaste automatizado.

4.3 Gestão Ágil

O Scrum é uma metodologia de gestão ágil que facilita alterações de projeto ao longo do seu desenvolvimento. Através de frequentes reuniões de grupo, a metodologia evidencia possíveis problemas na execução do projeto, e estimula discussões em busca de soluções rápidas.

Na plataforma Trello, os alunos encontram as tarefas designadas durante as reuniões para que sejam realizadas ao longo da semana. Lembretes de prazos de entrega também podem ser criados, notificando os usuários por e-mail ou aplicativo de celular.

Um exemplo de registro na plataforma Trello é apresentado na Figura 4. Nele estão divididas as tarefas distribuídas entre cada membro do Grupo E ao final do primeiro mês do projeto. Esses registros, realizados semanalmente

ou mensalmente, são importantes para verificar o desenvolvimento do trabalho em grupo e a conclusão de etapas.

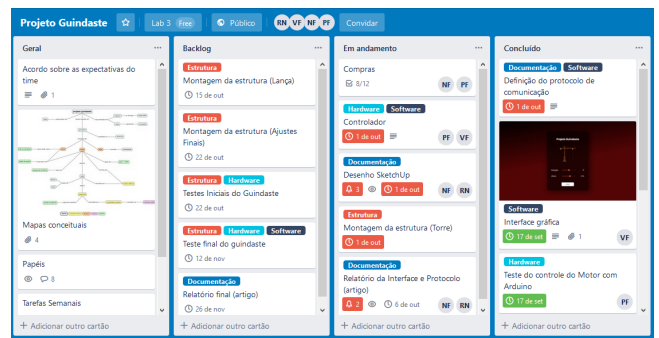


Figura 4. Registro de atividades via Trello do primeiro mês de projeto.

4.4 Modelo 3D

Respeitando a determinação de utilizar software gratuito, os alunos desenvolveram um modelo 3D no SketchUp da estrutura do guindaste automatizado. O programa permite criar um modelo realista da estrutura do guindaste, representando os diferentes tipos de palitos de picolé utilizados para a construção do protótipo. O modelo projetado pelo Grupo E pode ser visto na Figura 5.

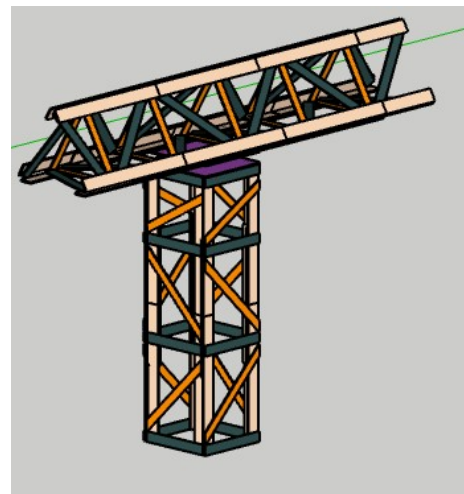


Figura 5. Modelo em SketchUp da estrutura do guindaste automatizado.

4.5 Projeto Eletroeletrônico

Atendendo um dos requisitos de projeto, o controlador no sistema é uma placa Arduino do modelo Mega 2560.

Para girar a lança do guindaste e mover o cabo preso ao eletroímã, alterando assim a altura deste em relação ao solo, o Grupo E optou por utilizar dois motores de passo do modelo 28BYJ-48. Cada motor é controlado através de um módulo ULN2003, com drivers para acionamento resultando num posicionamento preciso do atuador. Para tal, quatro saídas PWM do Arduino são conectadas em cada um desses módulos. Uma fonte de 5V capaz de fornecer 1A foi utilizada para a alimentação dos motores.

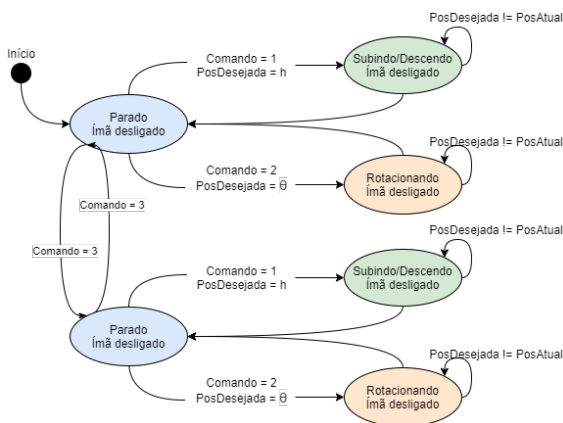


Figura 3. Diagrama de estados do guindaste automatizado.

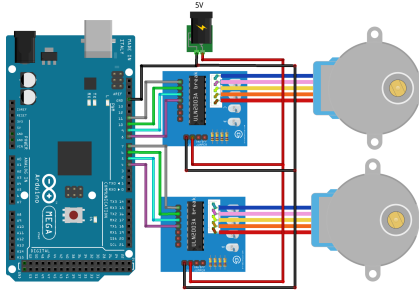


Figura 6. Esquemático de conexão entre Arduino e motores de passo com módulos ULN2003 e fonte de 5V.

A conexão entre o Arduino, módulos, motores e fonte é ilustrada na Figura 6.

Já para controlar o eletroímã, uma saída digital do Arduino foi conectada ao circuito de acionamento deste. Este circuito, mostrado na Figura 7, tem como principal característica um transistor, do modelo BC337, atuando em modo de corte ou de saturação. O diodo em paralelo ao eletroímã tem a função de impedir uma corrente de retorno quando o eletroímã é desativado, protegendo assim os demais componentes do circuito. Considerando a resistência interna de 50Ω e corrente de operação ($I = i_{Csat}$) de aproximadamente $250mA$ do eletroímã utilizado pelo Grupo E, é possível calcular o valor do resistor R_1 através das Equações 1, 2 e 3, indicando a utilização de um resistor de $1,8k\Omega$.

$$i_{Csat} = \frac{V_{CC} - V_{CEsat}}{R_{eletroima}} = \frac{12V - 0,2V}{50\Omega} = 236mA \quad (1)$$

$$i_{Bmin} = \frac{i_{Csat}}{\beta_{min}} = \frac{236mA}{100} = 2,36mA \quad (2)$$

$$R_1 = \frac{5V - V_{BE}}{i_{Bmin}} = \frac{5V - 0,7V}{2,36mA} = 1,822k\Omega \quad (3)$$

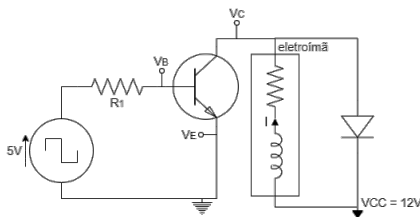


Figura 7. Circuito de acionamento do eletroímã.

4.6 Interface Gráfica de Usuário

Neste projeto, a interface gráfica permite ao usuário estabelecer os estados dos motores que controlam a lança do guindaste e a altura do eletroímã, além do estado ligado e desligado do eletroímã. A solução proposta pelo Grupo E foi a utilização de um ambiente *web*.

O *front-end* da interface foi implementada utilizando a quinta versão do *framework* Angular³. Essa escolha foi realizada com base na flexibilidade e capacidade de desenvolver aplicações complexas num tempo relativamente

³ <https://angular.io/>

curto e com qualidade. Na interface gráfica apresentada na Figura 8, é possível executar as seguintes ações:

- rotacionar a lança do guindaste: nesta seção, o usuário pode definir o ângulo de rotação do guindaste com valores de -180° até 180° , via *slider* ou *input*, com precisão de 1° . Enquanto a ação é executada, a interface fica bloqueada conforme planejado no diagrama de estados, sendo liberada apenas após uma confirmação do controlador ser recebida.
- definir a altura do eletroímã: a altura do eletroímã pode ser ajustada de $0cm$ até $20cm$, via *slider* ou *input*, com precisão de $1cm$. Esta ação também bloqueia a interface.
- ativar/desativar o eletroímã: por fim, o usuário pode ativar ou desativar o eletroímã, desde que este não esteja em movimento.

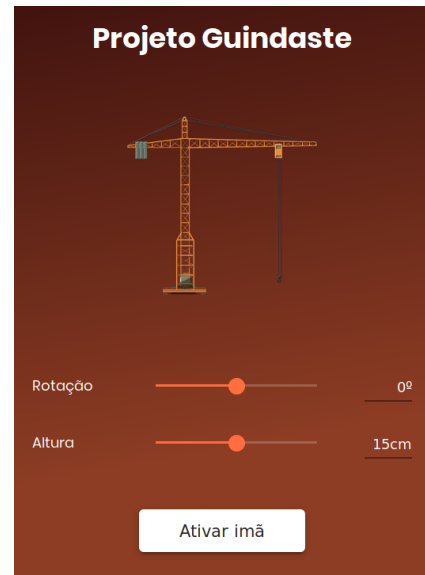


Figura 8. Interface gráfica de usuário.

Já o *back-end* do Grupo E foi implementado em Node.js⁴, um interpretador javascript de também de código-fonte aberto. O interpretador utilizado foi uma escolha de projeto, baseado em familiaridade dos alunos e facilidade de aplicação. A função do *back-end* neste sistema é comunicar serialmente com o controlador. O envio de mensagens entre *front-end* e *back-end* foi estabelecido com a biblioteca socket.io, que permite uma comunicação bidirecional entre o cliente e servidor.

4.7 Software Embarcado

Para o realizar o controle do guindaste, o Grupo E desenvolveu um software embarcado num microcontrolador Arduino Mega 2560.

O funcionamento básico do software embarcado é apresentado pelo pseudocódigo do Algoritmo 1, onde o controlador permanece em estado de espera (*loop*), até receber um comando do *back-end*. Após receber o comando, a ação a ser tomada é identificada e executada; o resultado é enviado de volta ao *back-end*, informando se esta foi executada com sucesso ou não, além de reportar o estado atual do guindaste.

⁴ <https://nodejs.org>

Algoritmo 1 Software Embarcado - Loop Principal

```
while Loop do
  if comando recebido then
    bytes ← lerBytes()
    acao ← obterAcao(bytes)
    valor ← obterValor(bytes)
    resultado ← executarAcao(acao, valor)
    enviarResposta(resultado)
```

Quanto às ações de controle executadas, são elas: inicializar o sistema, rotacionar a lança do guindaste, alterar a altura do eletroímã e ativar ou desativar o eletroímã.

A ação de inicializar o sistema é requisitada quando a interface gráfica é ativada, informando qual o estado atual do guindaste. Para tal, o software realiza a leitura do ângulo de rotação e da altura, salvos na memória EEPROM do Arduino, e depois os envia ao *back-end*.

Para rotacionar a lança do guindaste, é necessário que o controlador acione o motor de passo responsável por esta ação, através de 4 saídas PWM. É considerado que para rotacionar a lança em 1°, são necessários 1/360 vezes o número total de passos para uma revolução do motor, que é igual a 2038. Após comandar o motor para realizar o número calculado de passos, o novo ângulo é salvo na memória EEPROM do Arduino.

A ação de alterar a altura do eletroímã ocorre de forma análoga à rotação, porém acionando o motor responsável por liberar e retrair o cabo onde o eletroímã está fixado. Neste caso, para alterar esta altura em 1cm, são necessários 1/3.8 vezes o número total de passos para uma revolução do motor, que também é igual a 2038.

Já para ativar ou desativar o eletroímã, o controlador utiliza uma saída digital do Arduino.

4.8 Protocolo de Comunicação

A interação entre o *back-end* e o software embarcado no Arduino é realizada através de uma porta serial, que permite uma comunicação bidirecional conforme o protocolo de comunicação descrito nessa seção. Um dos requisitos de projeto imposto pela disciplina foi a troca de mensagens com tamanho máximo de 2 bytes.

A mensagem padrão utilizada pelo Grupo E está ilustrada na Figura 9. O primeiro byte é o de comando, definido através de quatro bits. Os dois bits menos significativos determinam qual ação será realizada; o terceiro indica se o valor de ação, fornecido no segundo byte, é negativo ou positivo. Já o quarto bit é utilizado apenas quando a mensagem parte do software embarcado para o *back-end*, indicando se houve erro na ação executada.

Já o segundo byte registra o valor relacionado à ação escolhida. Para o comando de rotação, ele pode representar valores entre 0 e 180, com resolução 1°; para a altura, valores entre 0 e 20, com resolução de 1cm; e para o eletroímã, 0 para ativá-lo ou 1 para desativá-lo. As mensagens enviadas do controlador para o *back-end* também seguem este protocolo. Assim, ao finalizar uma tarefa, o Arduino envia dois bytes contendo a ação realizada, o valor final registrado e se esta foi executada com sucesso ou não.

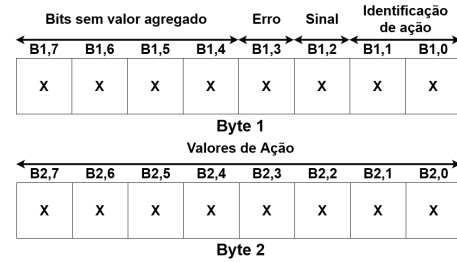


Figura 9. Distribuição de bits do protocolo de comunicação.

4.9 Integração

Ao longo da disciplina, os alunos implementam separadamente cada uma das partes do guindaste automatizado para integrá-las ao final do projeto. Os componentes envolvidos neste processo de integração estão destacados na Figura 10.

A etapa de integração é uma das mais desafiadoras durante a implementação do protótipo funcional. De maneira geral, a parte mecânica é a que os alunos de ES encontram maiores dificuldades, devido à menor afinidade se comparada às outras áreas abordadas no projeto.

Esse foi o caso do Grupo E, que teve dificuldades na fixação dos motores à estrutura do guindaste utilizando mancais, e também na passagem de fios de forma a não interferir na rotação e equilíbrio da lança. Após conectar o sistema eletroeletrônico e adicionar o eletroímã, os alunos precisaram ajustar a velocidade de rotação do motor da lança do guindaste, para que todo o conjunto permaneça em equilíbrio durante essa ação, e também refinar a relação entre uma volta do outro motor e a diferença de altura do eletroímã.

Para garantir a integridade das mensagens trocadas, foi necessário realizar um teste de integração entre o Arduino e a interface gráfica antes de conectá-los aos demais componentes do sistema. Por fim, para inicializar o sistema, os valores iniciais do ângulo de rotação da lança e altura do eletroímã devem ser salvos na memória do Arduino.

O protótipo final implementado pelo Grupo E é ilustrado na Figura 11. Além disso, um vídeo demonstrando o guindaste automatizado está disponível online⁵.

5. CONCLUSÃO

Este artigo apresentou uma metodologia de ensino de Engenharia de Sistemas aplicada aos alunos da UFMG através da disciplina Laboratório de Projeto III. Ao longo da disciplina, os alunos desenvolvem um projeto não trivial desde sua concepção até a implementação de um protótipo funcional, utilizando diferentes ferramentas de modelagem, concepção e gerenciamento de projetos.

Partindo de um caso de estudo definido a cada semestre, os alunos fazem um levantamento dos requisitos funcionais e não funcionais para a implementação do sistema. Através da elaboração de mapas conceituais, eles obtêm uma visão holística do projeto antes de iniciar sua construção. Para modelar o sistema através de estados e transições, os

⁵ <https://youtu.be/yrjsYbbNFQI>

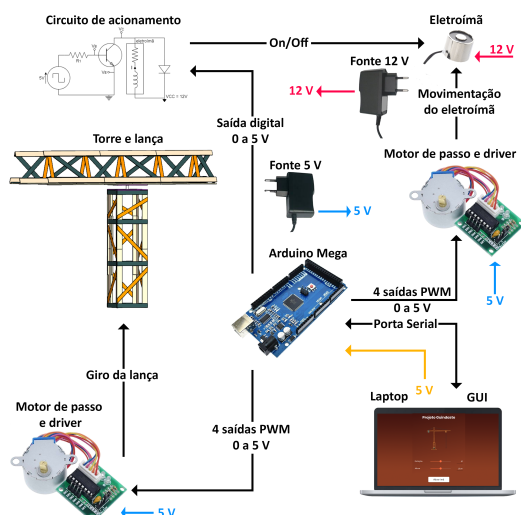


Figura 10. Esquemático de componentes que compõem o guindaste automatizado.

alunos utilizam um diagrama de Máquina de Estados Finitos (FSM). O gerenciamento do projeto é registrado na plataforma online Trello, aplicando as metodologias Scrum e Kanban de gestão ágil.

O caso de estudo proposto para os alunos do segundo semestre de 2019 foi a construção de um guindaste automatizado. Para este projeto específico, os alunos realizaram o projeto mecânico e construção da estrutura do guindaste utilizando palitos de picolé, implementação de circuitos e integração com microcontrolador para acionamento de motores e eletroímã, definição do protocolo de comunicação e desenvolvimento de interface gráfica de usuário.

A disciplina possibilita aos alunos aplicar na prática alguns dos conhecimentos e teorias estudadas ao longo do curso de Engenharia de Sistemas, fornecendo uma experiência real de integração de sistemas não triviais. A execução do projeto, incluindo gestão de tempo e divisão de tarefas, exige que os alunos desenvolvam habilidades interpessoais e superem desafios de trabalhar em grupo. Por fim, a construção de um protótipo em escala de baixo custo, utilizando hardware e software livres, viabiliza o emprego da metodologia didática proposta neste artigo em outras universidades e instituições de ensino públicas do País.

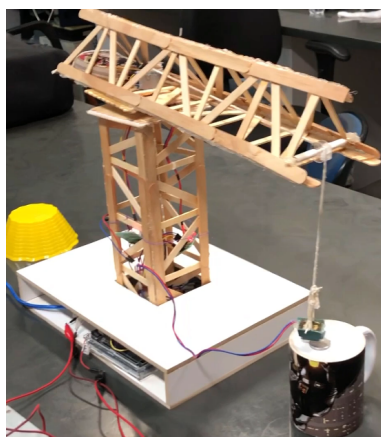


Figura 11. Guindaste automatizado construído pelos alunos do grupo E.

O acompanhamento do desenvolvimento dos projetos ao longo dos semestres mostra que os alunos enfrentam maiores dificuldades na construção e integração de hardware. Neste sentido, explorar o conteúdo produzido pelos grupos de semestres anteriores consiste em uma forma de auxiliar e guiar os alunos a lidar com tais dificuldades.

Para os próximos semestres, novos casos de estudo serão propostos. Inicialmente está planejado o desenvolvimento de um guindaste completamente autônomo, incluindo no hardware do protótipo um sensor de ultrassom. Em sequência, os alunos de Laboratório de Projeto III deverão implementar uma mini-empilhadeira autônoma. Um desafio futuro consiste em adaptar a metodologia proposta ao ensino remoto, garantindo o distanciamento social sem prejudicar o caráter prático da disciplina.

REFERÊNCIAS

- Abdel-Rahman, E.M., Nayfeh, A.H., and Masoud, Z.N. (2003). Dynamics and control of cranes: A review. *Journal of Vibration and Control*, 9(7), 863–908.
- Bell, S. (2010). Project-based learning for the 21st century: Skills for the future. *The Clearing House: A Journal of Educational Strategies, Issues and Ideas*, 83(2), 39–43.
- Denis, B. and Hubert, S. (2001). Collaborative learning in an educational robotics environment. *Computers in Human Behavior*, 17(5-6), 465–480.
- Faisandier, A. (2013). *Systems opportunities and requirements*. Sinergy'Com, 1 edition.
- Jezernik, K., Horvat, R., and Harnik, J. (2012). Finite-state machine motion controller: Servo drives. *IEEE Industrial Electronics Magazine*, 6(3), 13–23.
- Krehbiel, T.C., Salzarulo, P.A., Cosmah, M.L., Forren, J., Gannod, G., Havelka, D., Hulshult, A.R., and Merhout, J. (2017). Agile manifesto for teaching and learning. *Journal of Effective Teaching*, 17(2).
- Maruping, L.M., Venkatesh, V., and Agarwal, R. (2009). A control theory perspective on agile methodology use and changing user requirements. *Information Systems Research*, 20(3), 377–399.
- Mills, J. and Treagust, D. (2003). Engineering education, is problem-based or project-based learning the answer. *Aust J Eng Educ*, 3.
- Novak, J.D. (2002). Meaningful learning: The essential factor for conceptual change in limited or inappropriate propositional hierarchies leading to empowerment of learners. *Science Education*, 86(4), 548–571.
- Nussey, J. (2018). *Arduino for Dummies*. John Wiley & Sons, 2 edition.
- Piaget, J., Inhelder, B., Langdon, F.J., and Lunzer, J.L. (2006). *The child's conception of space*. W. W. Norton Company.
- Ramos, A.L., Ferreira, J.V., and Barcelo, J. (2012). Model-based systems engineering: An emerging approach for modern systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(1), 101–111.
- Santos, E.P., Guimarães, F.G., Adriano, R., Magalhães, A.L.C.C., Moro, L., and Mortimer, E.F. (2013). Laboratórios de projetos da engenharia de sistemas da ufmg: Aprender em novo contexto. *XLI Congresso Brasileiro de Educação em Engenharia, COBENGE, Gramado*.
- Sommerville, I. (2005). Integrated requirements engineering: a tutorial. *IEEE Software*, 22(1), 16–23.