



UNIVERSIDADE FEDERAL RURAL DO RIO DE JANEIRO
INSTITUTO DE FLORESTAS
CURSO DE GRADUAÇÃO EM ENGENHARIA FLORESTAL

FLÁVIA DA SILVA LEMOS

**MODELO CONCEITUAL PARA AUTOMAÇÃO DO FLUXO DE RESULTADOS NO
PLANEJAMENTO FLORESTAL**

Prof. Dr. SAMUEL DE PADUA CHAVES E CARVALHO
Orientador

SEROPÉDICA, RJ
JULHO – 2025



UNIVERSIDADE FEDERAL RURAL DO RIO DE JANEIRO
INSTITUTO DE FLORESTAS
CURSO DE GRADUAÇÃO EM ENGENHARIA FLORESTAL

FLÁVIA DA SILVA LEMOS

**MODELO CONCEITUAL PARA AUTOMAÇÃO DO FLUXO DE RESULTADOS NO
PLANEJAMENTO FLORESTAL**

Monografia apresentada ao Curso de Engenharia Florestal, como requisito parcial para a obtenção do Título de Engenheiro Florestal, Instituto de Florestas da Universidade Federal Rural do Rio de Janeiro.

Prof. Dr. SAMUEL DE PADUA CHAVES E CARVALHO
Orientador

SEROPÉDICA, RJ
JULHO – 2025



HOMOLOGAÇÃO Nº 18 / 2025 - DeptSil (12.28.01.00.00.00.00.31)

Nº do Protocolo: 23083.037224/2025-66

Seropédica-RJ, 10 de julho de 2025.

**MODELO CONCEITUAL PARA AUTOMAÇÃO DO FLUXO DE RESULTADOS NO PLANEJAMENTO
FLORESTAL**

FLÁVIA DA SILVA LEMOS

APROVADA EM: 09/07/2025

BANCA EXAMINADORA:

Prof. Dr. Samuel de Padua Chaves e Carvalho - UFRRJ (Orientador)

Prof. Dr. Marco Antonio Monte - UFRRJ (Membro)

Prof. Dr. Emanuel Jose Gomes de Araujo - UFRRJ (Membro)

(Assinado digitalmente em 11/07/2025 14:54)
**EMANUEL JOSE GOMES DE ARAUJO
PROFESSOR DO MAGISTERIO SUPERIOR
DeptSil (12.28.01.00.00.00.00.31)
Matrícula: 1978275**

(Assinado digitalmente em 10/07/2025 21:30)
**MARCO ANTONIO MONTE
PROFESSOR DO MAGISTERIO SUPERIOR
DeptSil (12.28.01.00.00.00.00.31)
Matrícula: 1972555**

(Assinado digitalmente em 10/07/2025 21:27)
**SAMUEL DE PADUA CHAVES E CARVALHO
PROFESSOR DO MAGISTERIO SUPERIOR
DeptSil (12.28.01.00.00.00.00.31)
Matrícula: 1115836**

Visualize o documento original em <https://sipac.ufrrj.br/public/documentos/index.jsp> informando seu número: **18**, ano: **2025**, tipo: **HOMOLOGAÇÃO**, data de emissão: **10/07/2025** e o código de verificação: **b65a891743**

AGRADECIMENTOS

Ao olhar pra trás e pensar na minha trajetória até aqui, me sinto mais do que grata pela contribuição de tantas pessoas que me acompanharam e espaços que fizeram parte desse caminho.

Agradeço à Deus por ter me acompanhar em toda minha trajetória.

À universidade agradeço por ter sido um espaço de descobertas, desafios e muito crescimento pessoal. Um agradecimento especial ao setor de auxílios estudantis, que foi fundamental para que eu pudesse me manter no curso e concluir o ensino superior.

Sou grata ao meu orientador, Samuel Carvalho, por confiar no meu potencial e apoiar minhas ideias, mesmo quando elas ainda pareciam um pouco fora do comum.

Ao meu tutor de estágio, Maykon Gabriel, obrigada por me apresentar e incentivar a explorar a linguagem R — uma ferramenta que trouxe autonomia e ampliou minha visão técnica, colaborando na minha formação profissional. Deixo também meu carinho à comunidade R e ao mundo *opensource*, que compartilham conhecimento de forma generosa e tornam o aprendizado mais acessível.

Aos bons professores e, com carinho especial, aos membros da banca que alimentaram meu saber com ética, dedicação e entusiasmo, deixo meu reconhecimento e respeito.

E, finalmente, agradeço aos meus pais, pelo apoio, incentivos e escolhas que fizeram para que eu pudesse chegar até aqui. Obrigada por acreditarem em mim e respeitarem meus interesses — graças a vocês, sigo com leveza, curiosidade e vontade de continuar aprendendo.

“Os dados são apenas resumos de milhares de histórias – conte algumas dessas histórias para ajudar a tornar os dados significativos.”
Chip & Dan Heath

RESUMO

Este estudo apresenta o desenvolvimento de um modelo conceitual destinado à automação do fluxo de resultados no planejamento florestal, com o objetivo de alavancar eficiência, reproduutibilidade e organização na análise de dados deste setor. A partir da linguagem R e do pacote Shiny, foi produzido um aplicativo interativo para visualizar indicadores florestais, usando dados simulados de operações reais. O modelo abrange desde a organização dos scripts até a geração de gráficos dinâmicos para apoio à tomada de decisão. A estrutura do sistema foi documentada por diagramas, com auxílio do *software* Draw.io, e todo o projeto foi versionado e publicado via GitHub, demonstrando o caráter aberto e replicável. Os resultados mostram a capacidade de integrar análise, automação e visualização interativa como ferramenta de apoio à gestão florestal moderna.

Palavras-chave: Visualização de dados. Reprodutibilidade analítica. Aplicações Web.

ABSTRACT

This study presents the development of a conceptual model aimed at automating the flow of results in forest planning, with the goal of enhancing efficiency, reproducibility, and organization in data analysis in this the sector. Using the R programming language and the Shiny package, an interactive application was developed to visualize forest indicators, based on simulated data from real operations. The model encompasses everything from the organization of scripts to the generation of dynamic charts to support decision-making. The system architecture was documented through diagrams using Draw.io, and the entire project was versioned and published via GitHub, demonstrating its open and replicable nature. The results highlight the potential of integrating analysis, automation, and interactive visualization as a tool to support modern forest management.

Keywords: Data visualization. Analytical reproducibility. Web applications.

SUMÁRIO

LISTA DE FIGURAS	viii
1. INTRODUÇÃO	1
2. REVISÃO DE LITERATURA	1
2.1. Planejamento Florestal	1
2.2. A linguagem R e sua aplicação no processamento de dados florestais	2
2.3. Controle de versão e reproduzibilidade com GitHub.....	4
2.4. Comunicação de resultados com Shiny	5
3. MATERIAL E MÉTODOS	5
3.1. Caracterização da pesquisa.....	5
3.2. Base de dados	6
3.3. Recursos Utilizados	7
3.3.1. Ferramentas utilizadas	7
3.3.2. Pacotes R utilizados.....	8
3.4. Estruturação do modelo conceitual.....	9
3.5. Geração automatizada de relatórios.....	10
3.6. Repositório GitHub e <code>renv</code> como ferramentas de gestão do projeto	11
4. RESULTADOS E DISCUSSÃO	13
5. CONCLUSÃO	22
6. REFERÊNCIAS BIBLIOGRÁFICAS.....	23

LISTA DE FIGURAS

Figura 1: Detalhamento de horizontes do planejamento.....	2
Figura 2: Fluxograma de trabalho	10
Figura 3: Scripts individuais para cada gráfico.....	12
Figura 4: Momentos básicos de aplicação das funções do pacote renv.	13
Figura 5: Fluxo de estruturação dos resultados.....	15
Figura 6: QR Code para abrir o aplicativo em HTML.....	16
Figura 7: Download do projeto via site GitHub através dos botões Código e Baixar ZIP.	16
Figura 8: Botão “Run App” para abrir o aplicativo em HTML.	17
Figura 9: Página 1 do aplicativo: Instruções de uso.....	17
Figura 10: Página 2 do aplicativo: Upload do arquivo.	18
Figura 11: Página 2 do aplicativo: Pré visualização dos dados.....	18
Figura 12: Página 5 do aplicativo: Idade de Colheita (anos) e destaque para slider.....	19
Figura 13: Página 7 do aplicativo: Colheita Jovem (mil m ³).	19
Figura 14: Página 3 do aplicativo: Volume de Eucalipto (mil m ³) e destaque para uma classe.	20
Figura 15: Página 4 do aplicativo: Faturamento (mi R\$).....	20
Figura 16: Página 6 do aplicativo: Distância Média de Transporte (km).	21
Figura 17: Alterações de scripts no Github onde o grifo em vermelho simboliza a exclusão da linha do script e o grifo em verde simboliza a adição.	22

1. INTRODUÇÃO

De acordo com o Relatório Anual da Indústria Brasileira de Árvores (IBÁ, 2024), a importância do setor de árvores cultivadas na economia nacional é evidente, notadamente com uma receita bruta de R\$ 202,6 bilhões em 2023, e, ainda, com um aumento médio anual de 3,2% nos últimos dez anos. Essa ascensão reflete o crescimento da demanda por produtos florestais e o uso de boas estratégias eficazes em planejamento e gestão nas florestas plantadas.

O planejamento estratégico florestal, se revela, portanto, como um sistema que se destaca como necessário para o manejo sustentável dos recursos naturais, priorizando o uso eficiente dos ativos florestais a longo prazo. Angelo *et al.* (2017) ressaltam a relevância de sistemas de informação bem construídos como apoio à gestão florestal, apontando a produção e a economia florestal como áreas prioritárias para o desenvolvimento de indicadores e tomada de decisão.

No entanto, o panorama geral do planejamento florestal ainda apresenta impasses relacionados à gestão e integração dos dados. Em um setor que conta com mais de 10,4 milhões de hectares de florestas plantadas no Brasil, sendo o eucalipto representado por 76% e o pinus por 20% dessa área (IBÁ, 2024), as informações operacionais e estratégicas vêm através de diferentes setores e sistemas, como inventário, colheita, logística e planejamento industrial. Essa diversidade de fontes, formatos e objetivos traz fragmentação nos processos de análise, dificultando a padronização dos procedimentos, a rastreabilidade das alterações e a democratização dos resultados. Os desafios se encontram na construção de indicadores técnicos, frequentemente calculados de forma isolada ou repetitiva. A ausência de fluxos bem definidos e automatizados compromete a reproduzibilidade das análises, além de aumentar o tempo necessário para extração de informações úteis para tomada de decisões.

Diante do cenário apresentado, este trabalho tem como objetivo geral delinear um modelo conceitual para automação do fluxo de resultados no planejamento florestal, visando aprimorar o acesso e a análise dos dados de maneira organizada e eficiente. Em particular, buscou-se propor um modelo teórico capaz de estruturar fluxos que promovem maior transparência e reproduzibilidade nos processos analíticos mediante a geração de gráficos.

2. REVISÃO DE LITERATURA

2.1. Planejamento Florestal

O setor de planejamento florestal organiza e coordena o uso dos recursos florestais ao longo do tempo, balanceando produtividade, sustentabilidade e estimativas financeiras. Esse planejamento envolve atividades como definir áreas de plantio, colheita, manejo e logística, baseado em estudos técnicos e projeções do crescimento e da produção florestal. Ele age em diversos horizontes temporais, de curto, médio, longo prazo e operacional (Figura 1). O planejamento é também crucial para assegurar, principalmente, que a produção de madeira ocorra de forma constante, proveitosa e alinhada com os limites legais, ambientais e exigências de consumo de fábrica (Kaczmarek, 2021). Ademais, de acordo com a Organização das Nações Unidas para Alimentação e Agricultura (FAO, 2003), bons planos florestais devem conter cinco características essenciais: ligação com políticas maiores, visão das oportunidades e consequências futuras, abordagem estratégica, execução e avaliação regular dos resultados.

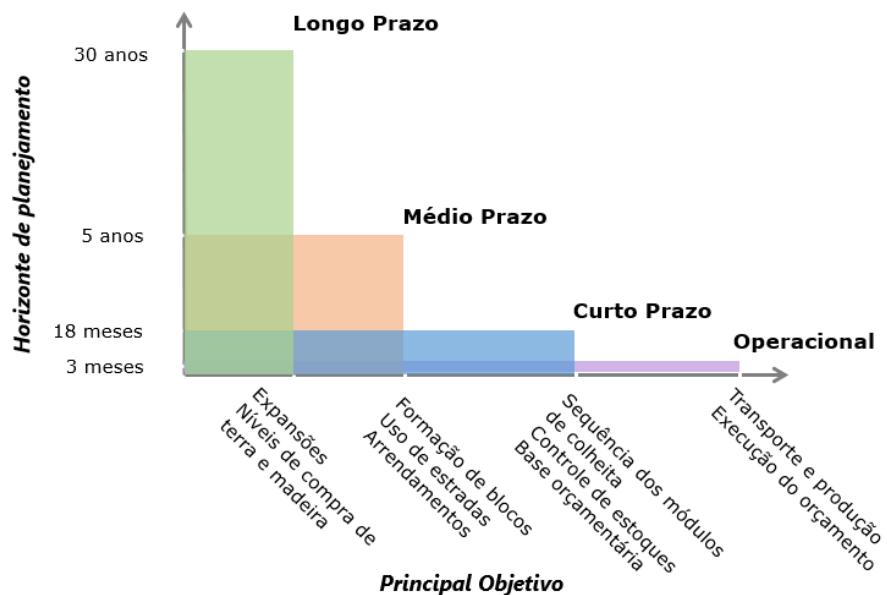


Figura 1: Detalhamento de horizontes do planejamento. Fonte: Adaptado de Kaczmarek (2021).

Nesse contexto, a criação dos planos depende da integração de diferentes modelos de dados, manipulados por vários sistemas da empresa, cada um com sua própria estrutura e suas peculiaridades, o que pode dificultar a consistência e agilidade das análises, base para tomada de decisões do planejamento florestal (Oliveira Filho *et al.*, 2003).

2.2. A linguagem R e sua aplicação no processamento de dados florestais

A linguagem R é um ambiente estatístico e também de programação, amplamente utilizado para análise de dados, modelagem estatística e visualização gráfica, que pode ser instalado para os sistemas operacionais Windows, macOS e Linus diretamente no site oficial da linguagem: <https://cran.r-project.org/>. Nascida com código aberto, R oferece um grande ecossistema de pacotes disponíveis no CRAN (Comprehensive R Archive Network), que permitem a expansão de suas funcionalidades para as mais diversas áreas, incluindo *machine learning* e sintaxe otimizada para manipulação e exploração de dados (R Foundation, 2024).

Empresas de papel e celulose, por exemplo, frequentemente lidam com volumes de dados extensos e heterogêneos como inventários florestais, sensoriamento, mensurações e dados logísticos, que facilmente ultrapassam as capacidades de muitos dos *softwares* disponíveis atualmente. Nesse contexto, o R se destaca por sua capacidade de oferecer ferramentas abertas para análise, modelagem, visualização e tabulação de dados, com mais de 100 pacotes voltados para o setor florestal (Atkins *et al.*, 2022). Além disso, é amplamente empregado por quase metade dos estudos florestais recentes, conforme levantamento de 14.800 artigos, o R oferece suporte a modelagens estatísticas sofisticadas como regressões mistas e análise de diversidade (Lai *et al.*, 2023), extrapolando o simples processamento de dados e constituindo-se em ferramenta essencial para análise e visualização de dados.

A eficácia do R é aprofundada quando usado dentro do RStudio, um ambiente de desenvolvimento integrado (IDE) que centraliza scripts, console, plots, histórico e gestão de pacotes em uma mesma interface que pode ser obtido através do site: <https://posit.co/download/rstudio-desktop/#download>. O RStudio melhora substancialmente a produtividade ao permitir navegação facilitada entre projetos, autocompletar objetos e organizar diretórios de forma mais eficiente, especialmente em pipelines automatizados (Posit, 2023). Sua compatibilidade com R Markdown permite ainda a criação de documentos dinâmicos que integram código, imagens, escrituras, resultados e interpretações em um único relatório, o que amplia a transparência e a repetibilidade que podem ser usadas nas análises conduzidas por equipes técnicas em diferentes etapas do planejamento florestal (Baumer *et al.*, 2014).

A linguagem R tem instrumentos para se unir com outras soluções e linguagens, acelerando seu uso em diagnósticos mais complexos. Ao incorporar LaTeX pelo R Markdown, por exemplo, há possibilidade de realizar a produção de documentos técnicos e acadêmicos bem

formatados, incrementando resultados, código e texto num único arquivo, o que auxilia na precisão da documentação (RStudio, 2014).

Pra melhorar ainda mais o desempenho, o pacote `rcpp` permite a inclusão de código C++ nos scripts R, o que é útil para acelerar cálculos robustos em modelagens e simulações florestais (Rcpp Core, 2025). A conexão com Python também pode ocorrer, com a utilização o pacote `reticulate`, que viabiliza a execução de scripts Python no R, aumentando as possibilidades de uso com a disponibilidade de mais bibliotecas, especialmente em *machine learning* (Posit, 2025).

Além disso, é possível obter uma plataforma Stan com o auxílio do pacote `rstan`, ele traz consigo uma interface que fornece assistência à criação de modelos estatísticos bayesianos. Essa abordagem se aplica à análise de incertezas e previsões, pois a modelagem probabilística pode auxiliar, de forma mais eficaz, na tomada de decisões (Carpenter, 2025).

2.3. Controle de versão e reproduzibilidade com GitHub

A documentação oficial do Git (GIT-SCM, 2024), diz que o sistema de controle de versão distribuído simplifica o trabalho colaborativo. Linus Torvalds, o criador, estruturou isso em repositórios para guardar e buscar as diferentes versões de um projeto, podendo estas serem recuperadas conforme a necessidade. A ferramenta é amplamente utilizada pela comunidade de desenvolvedores de *softwares*, programadores e na organização de projetos de ciência de dados, garantindo integridade e rastreabilidade das mudanças.

O GitHub, por sua vez, é uma plataforma baseada na web que hospeda repositórios Git e oferece funcionalidades para colaboração, controle de versão e gerenciamento de projetos. Além de permitir a integração contínua da sequência de tarefas, o GitHub oferece ferramentas como GitHub Actions, que possibilita a execução de etapas automatizadas sempre que algo for alterado no repositório, como um *push* e *pull request* aprimorando a eficiência dos projetos (GitHub, 2024). A combinação do uso de R com Git e GitHub é adotada para garantir a reproduzibilidade e versionamento de análises, possibilitando que equipes colaborem de maneira organizada e segura na manipulação de dados, utilização de ferramentas e no desenvolvimento de modelos analíticos.

Para garantir tais funções, o Git registra detalhadamente cada alteração realizada nos arquivos, incluindo informações sobre o autor, o momento da modificação e a descrição das

mudanças, permitindo a reconstrução exata do projeto em qualquer estádio anterior. Essa capacidade de versionamento facilita o rastreamento das evoluções, evita conflitos entre diferentes versões e possibilita a reversão rápida a estados estáveis. No ambiente colaborativo do GitHub, recursos como *branches* e *pull requests* promovem a organização do trabalho em paralelo e a revisão criteriosa antes da integração das mudanças, assegurando que as análises e modelos desenvolvidos permaneçam consistentes, auditáveis e atualizados em equipe (Chacon e Straub, 2014; GitHub, 2024).

2.4. Comunicação de resultados com Shiny

O Shiny é um pacote R feito pela empresa Posit, os mesmos criadores do R Studio, para trocar códigos R em aplicativos web que interagem com o usuário. Assim não se precisa codificar em linguagens como HTML ou JavaScript. Ele trabalha com duas partes chave: a face do usuário (UI) e o servidor (server), integrados de tal forma a mudar os resultados conforme a interação do usuário com os dados. Com isso é possível criar painéis com botões, menus, páginas e gráficos dinâmicos que respondem automaticamente às escolhas feitas dentro do app (Posit, 2024a).

Como o Shiny foi desenvolvido singularmente para o R, ele traz consigo os mesmos pacotes usados na análise de dados dentro do ambiente R. Isso torna fácil a junção com pacotes como `ggplot2`, `dplyr`, `plotly`, `ggiraph` e `echarts4r`, dando ao usuário final uma maneira visual e interativa de examinar os resultados da análise sem ter que trocar o código. O layout das aplicações também é altamente personalizável, com diferentes opções de navegação, fontes, abas, painéis e organização da tela com os mais diversos temas, o que melhora a experiência de quem acessa o relatório (Posit, 2024b).

3. MATERIAL E MÉTODOS

3.1. Caracterização da pesquisa

A pesquisa realizada demonstra uma natureza descritiva e exploratória. Considera-se como base um sistema genérico de produção florestal, construído para representar padrões habituais de organização e gestão do setor de planejamento florestal. A proposta metodológica busca

estabelecer reproduzibilidade e clareza na criação do modelo conceitual, através de simulações controladas, ferramentas de controle de versão e isolamento do ambiente computacional, para assegurar a consistência da aplicação gerada e simplificar sua replicação em contextos semelhantes.

3.2. Base de dados

Os dados utilizados neste estudo foram gerados por meio de uma simulação utilizando o pacote `wakefield`, com a finalidade de se obter um cenário característico de povoamentos florestais destinados à produção de celulose e papel. A simulação considerou talhões plantados entre o ano de 2012 e 2024, considerando a média da produtividade nacional de 33,7 m³/ha/ano, com idade de corte aos 7,2 anos, conforme divulgado no relatório anual da Indústria Brasileira de Árvores (IBÁ, 2024). A partir disso, foi introduzida uma variação de incremento médio anual (IMA) entre 30,7 e 36,7 m³/ha/ano na tentativa de refletir a heterogeneidade entre os talhões, simulando diferentes condições de sítio.

Em seguida, foram definidos atributos como idade de colheita, ano de colheita, distância média de transporte, volume por talhão e modalidade de fornecimento de madeira (se é própria da empresa ou comprada), onde se assumiu que toda madeira proveniente das áreas próprias da empresa é certificada, enquanto a madeira de terceiros (modalidade "compra") representa até 30% do volume total anual e corresponde à quantidade permitida pelas certificadoras para misturas de fontes não certificadas. Essa proporção segue os critérios estabelecidos pelo sistema FSC (Forest Stewardship Council), que permite a composição de produtos com até 30% de madeira controlada, desde que atenda a critérios específicos de origem e rastreabilidade (FSC, 2024). Essa simplificação foi adotada para simular um exemplo real, possibilitando o controle da origem do volume em análises e conformidade com políticas de cadeia de custódia.

Com relação ao valor de mercado da madeira, levou-se em consideração o preço médio de madeira em pé R\$ 135,00/estere para fins de celulose (CEPEA, 2025). Para a conversão do volume de estere para metro cúbico, utilizou-se o fator de 0,725 m³/st definido na Portaria IEF-MG nº 159/2012, resultando em um valor de R\$ 97,87 por metro cúbico, com base nisso o valor de faturamento em R\$/m³/ano foi obtido ao multiplicar o volume em m³ do ano de interesse com o valor médio por metro cúbico (Minas Gerais, 2012).

A partir desses critérios a demanda anual de fábrica foi fixada em 100.000 m³ de madeira para abastecimento de uma pequena indústria de celulose que tem apenas 100 talhões. Prevê-se que cada hectare colhido, ao final de um ciclo de 7,2 anos, produzirá 242,64 m³/ha (33,7 m³/ha/ano multiplicado por 7,2 anos). Através dessa produção, são necessários cerca de 412,13 hectares colhidos anualmente para suprir a demanda (100.000 m³ de demanda de fábrica dividido por 242,64 m³/ha). Levando em conta um sistema produtivo em que toda a área colhida é replantada a cada ano, estima-se um fundo florestal de aproximadamente 2.967 hectares (412,13 hectares multiplicados por 7,2 anos), quantidade suficiente para garantir constância sob um regime rotativo completo, balanceando plantio e colheita anualmente de maneira equilibrada. Com base nesse valor adquirido, as áreas de cada talhão foram geradas de maneira aleatorizada de modo que o somatório das áreas atinja o valor necessário para suprir o abastecimento da fábrica.

A idade atual gerada foi oriunda do ano de plantio calculado em detrimento ao ano atual, onde caso a idade atual é menor ou igual a 5 anos, a idade de colheita será a própria informação da idade atual, e quando não ocorrer esse fato, o ano de colheita plotado é de 7 anos. Desse modo, utilizando o ano de colheita e o IMA variável, foi obtido o valor de volume em m³/ha, multiplicando a idade de colheita e com a IMA variável e também o volume em m³/talhão multiplicando o volume em m³/ha pela área dos talhões, assim considerando um regime de manejo de corte raso com replantio anual em regime rotativo completo.

As informações construídas são meramente ilustrativas e foram geradas unicamente para demonstrar a estrutura e o funcionamento do modelo proposto. Em uma aplicação prática, variáveis como dados dendrométricos detalhados, custos operacionais de colheita, transporte e custos de estoque influenciariam diretamente a configuração da base de dados e, consequentemente, os resultados nos gráficos. Entretanto, essas considerações não foram avaliadas nesta etapa, já que a prioridade é a criação e validação da lógica de automação e viabilização do modelo.

3.3. Recursos Utilizados

3.3.1. Ferramentas utilizadas

O avanço do trabalho e da construção do aplicativo demandou o uso conjunto de ferramentas auxiliares, cada uma com suas finalidades específicas no processo. A Tabela 1

apresenta um resumo das quais foram utilizadas na organização e construção do projeto desde o rascunho até o cumprimento dos objetivos do trabalho.

Tabela 1: Ferramentas utilizadas, link de suas principais informações e finalidades.

Pacote	Finalidade	Link oficial da ferramenta com autores
R	Manipulação, análise e visualização de dados	https://www.r-project.org/contributors.html?utm_source=chatgpt.com
RStudio	Ambiente de desenvolvimento integrado com interface que facilita a programação	https://posit.co/blog/rstudio-is-becoming-posit/
Draw.io	Diagramação	https://www.drawio.com/about
Git	Permite que múltiplos desenvolvedores colaborem em um mesmo projeto simultaneamente com controle de versão	https://git-scm.com/book/pt-br/v2/Come%C3%A7ando-Uma-Breve-Hist%C3%B3ria-Git
GitHub	Hospeda e compartilha repositórios Git Criação de conteúdo dinâmico com R e outras linguagens e sistema de publicação científica de conteúdo como HTML, PDF, Word, ePub, etc.	https://github.com/about
Quarto		https://quarto.org/about.html

Fonte: Dados originais da pesquisa. Autora (2025).

3.3.2. Pacotes R utilizados

Há um conjunto integrado de pacotes e ferramentas auxiliares da linguagem R, que inclui funções reutilizáveis com documentações que ditam como usá-las e possuem dados de exemplo que facilitam o entendimento para resolver problemas específicos. Diferentes pacotes R foram utilizados ao longo do projeto de acordo com a finalidade (Tabela 2).

Tabela 2: Pacotes R utilizados e suas finalidades.

Pacote	Finalidade	Link da base da comunidade R com autores
bslib	Personalização de temas em Shiny	https://cran.r-project.org/package=bslib
dplyr	Manipulação eficiente de dados	https://cran.r-project.org/package=dplyr
echarts4r	Gráficos interativos com Echarts	https://cran.r-project.org/package=echarts4r
forcats	Manipulação de variáveis categóricas	https://cran.r-project.org/package=forcats
here	Gerenciamento seguro de caminhos de arquivos	https://cran.r-project.org/package=here
lubridate	Manipulação e cálculo com datas e horários	https://cran.r-project.org/package=lubridate

purrr	Programação funcional com listas	https://cran.r-project.org/package=purrr
readxl	Leitura de arquivos .xlsx	https://cran.r-project.org/package=readxl
shiny	Criação de aplicações web	https://cran.r-project.org/package=shiny
tidyverse	Organização e transformação de dados	https://cran.r-project.org/package=tidyverse
wakefield	Geração de dados simulados	https://cran.r-project.org/package=wakefield
writexl	Exportação de dados para Excel	https://cran.r-project.org/package=writexl

Fonte: Dados originais da pesquisa. Autora (2025).

3.4. Estruturação do modelo conceitual

O modelo teórico visa representar, com clareza e facilidade, a estrutura lógica do aplicativo construído ao longo deste trabalho. Para a construção, empregou-se o *software* Draw.io, uma ferramenta que permite a criação de fluxogramas e esquemas visuais.

No fluxograma elaborado, cada bloco leva uma função importante do sistema, desde a entrada dos dados até a forma de exibir os gráficos usados na análise. Os scripts de gráficos possuem a *pipeline* de geração, que trata dos dados e cria o visual do gráfico com o pacote echarts4r. Já o script de código do aplicativo em shiny foi estruturado em três partes: (i) Load all, que inclui o carregamento das bibliotecas e das funções armazenadas na pasta R/; (ii) UI detalhado, que define a interface do aplicativo com painéis, botões, estilos e organização visual dos gráficos; e (iii) Server detalhado, onde acontecem as verificações dos dados, os filtros por ano, o processamento reativo e a renderização dos gráficos de forma interativa.

O fluxo completo apresentado no modelo teórico expõe a dinâmica do sistema, seguindo a seguinte sequência: entrada de dados, leitura e validação das informações, filtragem de dados, geração dos gráficos, e, por fim, análise através da visualização que embasarão decisões futuras. Essa representação teórica construída no Draw.io funciona, portanto, como um guia visual que documenta o funcionamento do sistema desenvolvido, um suporte metodológico para replicação de relatórios gráficos. A seguir, apresenta-se o fluxograma de estrutura do estudo (Figura 2), que ilustra de forma sequencial as etapas metodológicas desenvolvidas ao longo da pesquisa, desde o início das atividades até a finalização do modelo e aplicativo proposto.

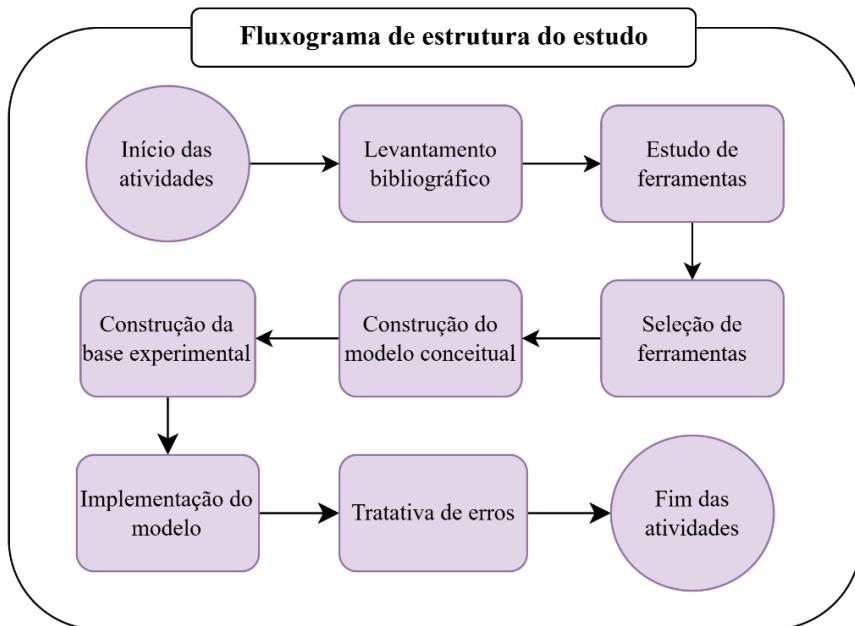


Figura 2: Fluxograma de trabalho. Fonte: Autora (2025).

3.5. Geração automatizada de relatórios

A construção do aplicativo para geração automatizada de relatórios foi feita em uma estrutura modular e organizada através do pacote `shiny`, com separação lógica entre os elementos de visualização, processamento e controle da aplicação, auxiliando na manutenção e futuras expansões da ferramenta.

As funções responsáveis por criar os gráficos foram armazenadas na pasta R, cada uma com sua definição da função gráfica, que inclui tanto o pré-processamento dos dados quanto a formatação do gráfico em si, operando com o pacote `echarts4r`. Esse formato modular possibilita que cada gráfico seja editado, testado ou reutilizado de forma independente.

Durante a construção da visualização interativa, foram avaliadas diferentes alternativas de pacotes em R. Inicialmente, foram exploradas abordagens utilizando o pacote `ggplot2` combinado com o pacote `plotly`, bem como o pacote `ggiraph`, que carregam consigo funções de *tooltips* interativas com os dados nos gráficos. No entanto, após testes e comparações, o pacote `echarts4r` mostrou-se mais adequado para os objetivos do projeto, oferecendo maior flexibilidade de personalização estética, integração fluida com o pacote `shiny`, leveza na renderização dos gráficos e um bom suporte a temas e estilos interativos.

O script central do aplicativo, que localizado na pasta 03_scripts do projeto, foi responsável por reunir todos os elementos do aplicativo desenvolvido no pacote shiny. Ele inicia com a seção “Load all”, onde são carregadas as bibliotecas principais (shiny, bslib, echarts4r, entre outras) e as funções gráficas armazenadas na pasta R, por meio da função walk disponível no pacote purrr, sobre a lista de arquivos .R.

A interface do usuário (UI) foi organizada por meio da função page_navbar(), com o uso do pacote bslib para personalização visual, incluindo o controle e aplicação de cores e fontes de todas as páginas e inclusão do R Markdown que contém as instruções de uso da ferramenta. Cada aba do aplicativo corresponde a um gráfico específico e inclui controles como sliderInput() para seleção do intervalo de anos. Já a lógica do servidor (server) inclui rotinas de validação dos dados de entrada; programações de avisos, a citar: "Envie um arquivo para visualizar o gráfico.", "Este gráfico requer colunas com as informações: Ano de colheita, Idade de Colheita e Volume" e "As colunas existem, mas a ausência de dados em uma ou mais delas impede a criação do gráfico. Verifique os dados."; aplicação de filtros conforme o intervalo selecionado, e a renderização dos gráficos a partir das funções previamente carregadas.

Somado a isso, o aplicativo inclui uma aba inicial com instruções para o usuário, implementadas em Markdown (instrucoes.Rmd), e recursos visuais como logotipos institucionais incorporados por meio de HTML, reforçando a identidade do projeto.

Essa organização em camadas, com as funções gráficas separadas na pasta R, o script do aplicativo centralizado na pasta 03_scripts, e os dados e recursos adicionais alocados em diretórios próprios, garante uma estrutura limpa, reutilizável e compatível com boas práticas de desenvolvimento em R, facilitando tanto o uso do aplicativo por terceiros quanto sua atualização e/ou contribuição futura.

3.6. Repositório GitHub e renv como ferramentas de gestão do projeto

O projeto foi gerenciado utilizando o sistema de controle de versão Git, com o código-fonte hospedado em um repositório no GitHub. Essa abordagem permitiu o armazenamento organizado de todos os scripts, dados e documentos relacionados a formação do aplicativo e das análises. A estrutura do repositório foi organizada em pastas específicas, incluindo diretórios para dados brutos, scripts das funções gráficas (pasta R), como demonstrado na

Figura 3, scripts do aplicativo shiny (pasta 03_scripts) e documentação, garantindo facilidade de navegação e manutenção.

Name	Size	Modified
graph_abastecimento.R	2.1 KB	Jun 22, 2025, 10:25 AM
graph_colheita_jovem.R	1.8 KB	Jun 22, 2025, 10:25 AM
graph_dmt.R	1.3 KB	Jun 22, 2025, 10:25 AM
graph_faturamento.R	1.1 KB	Jun 22, 2025, 10:24 AM
graph_idade_corte.R	925 B	Jun 22, 2025, 10:25 AM

Figura 3: Scripts individuais para cada gráfico. Fonte: Resultados originais da pesquisa.

Para controle e registro das alterações, há possibilidade da realização de *commits* frequentes com mensagens descriptivas, permitindo acompanhar o histórico da elaboração e facilitar a identificação de versões estáveis do projeto.

O repositório no GitHub serviu como base para a publicação do aplicativo, utilizando recursos de hospedagem como GitHub Pages ou serviços compatíveis para aplicações Shiny, possibilitando o acesso remoto à aplicação via link público ou restrito. O uso do GitHub também viabilizou a colaboração e o compartilhamento do projeto, facilitando a replicação dos procedimentos e a integração de contribuições externas quando necessário.

Foram incluídos no repositório o projeto como um todo, incluindo os códigos e o fluxograma construído no Draw.io, que representa visualmente todas as etapas do modelo teórico implementado, e esta monografia, o que facilita a compreensão completa do processo por parte de outras pessoas que queiram entender ou replicar a aplicação desenvolvida. Essa estrutura visa não apenas o armazenamento do projeto, mas também a sua acessibilidade e continuidade, permitindo que outras pessoas possam utilizar, adaptar ou evoluir o aplicativo.

Além da plataforma GitHub, foi usado o pacote `renv` pra gerir o ambiente de trabalho dentro do projeto em R. Essa ferramenta permite isolar as versões dos pacotes utilizados, criando um ambiente único para o projeto e evitando conflitos com futuras atualizações (RSTUDIO, 2025). Desta forma, garante-se a execução consistente do código ao longo do tempo e em diferentes máquinas. Para garantir que os pacotes estarão na mesma versão para trabalho para todos que acessarem o projeto, feito o download via GitHub basta executar a função `restore` do `renv` para sincronizar. A seguir na Figura 4, o esquema detalha os passos para gerenciar as versões dos pacotes a fim de gerar uma documentação e treinar os usuários para o devido uso.

```

Exemplo da necessidade do restore.
O projeto está fora da sincronia.
~ The project is out-of-sync -- use `renv::status()` for details.
[print] masked print.data.frame
Error in file.exists(pathname) :
  problema na conversão do nome de arquivo -- nome muito longo?
> targets::tar_make()
> targets::tar_make()
The project is out-of-sync -- use `renv::status()` for details.
[print] masked print.data.frame
The following package(s) are out of sync [lockfile != library]:
# CRAN -----
- lattice [0.22-6 <- 0.22-5]
- MASS [7.3-0@ 0.1 <- 7.3-0@]
See ?renv::status() for advice on resolving these issues.

Exemplo da necessidade do restore.
O projeto está configurado para uma versão do pacote específica
porém, a versão presente na biblioteca é outra, é preciso baixar a
versão especificada na configuração do renv.
Restarting R session ...
renv 1.0.5 was loaded from project library, but this project is configured to use renv 1.0.7.
- Use 'renv::record("renv@1.0.5")' to record renv 1.0.5 in the lockfile.
- Use 'renv::restore(packages = "renv")' to install renv 1.0.7 into the project library.
- Project 'C:/Users/EAFL00157132/Desktop/Git - Projetos R/20240610_Inputs_Woodstock_PR_2024_2' loaded. [renv 1.0.5]
- The project is out-of-sync -- use `renv::status()` for details.

Exemplo da necessidade do snapshot.
Os pacotes estão no estado inconsistente. É preciso tirar um "print
da situação" atual para que esses estados estejam não só baixados
mas também recordados no renv.
> renv::status()
The following package(s) are in an inconsistent state:
data.frame [2, 4]
package   chr forcats foreign
installed  chr y
recorded  chr n
used      chr y
See ?renv::status() for advice on resolving these issues.

Exemplo da necessidade do restore ou install.packages(c("dplyr",
"janitor", "readxl", "writexl"))
O projeto está usando os pacotes mas não estão instalados no ambiente renv.
Lockfile written to "C:/Users/EAFL00157132/OneDrive - Klabin/Silvicultura/renv.lock".
> renv::status()
The following package(s) are used in this project, but are not installed:
- dplyr
- janitor
- readxl
- writexl
See ?renv::status() for advice on resolving these issues.
> renv::restore()

renv::status para verificar se o projeto está consistente na questão dos pacotes
renv::restore para instalar pacotes listados no arquivo renv.lock, registrados no lockfile, na biblioteca do projeto
renv::install para instalar pacotes já dentro do renv caso algum falte - o mesmo que ir em packages
renv::snapshot para capturar o estado do projeto com renv

Após o restore ou snapshot restauramos a seção e novamente verificamos o status.
Quando esse aviso for dado quer dizer que estamos com pacotes consistentes no projeto.
> renv::status()
No issues found -- the project is in a consistent state.
>

```

Figura 4: Momentos básicos de aplicação das funções do pacote `renv`. Fonte: Autora (2025).

4. RESULTADOS E DISCUSSÃO

O aplicativo construído juntou as informações simuladas, ofertando uma interface visual interativa, organizada em abas temáticas separadas, mostrando os principais indicadores do planejamento florestal, o volume em mil m³, o faturamento em mi R\$, a idade de colheita em anos, a distância média de transporte em km e a colheita jovem realizada em mil m³. Através da análise dos scripts implementados, foi possível mostrar a sequência das fases englobadas na geração do sistema.

Foi construído a partir do modelo conceitual (Figura 5), onde se apresenta o fluxo de dados construído no Draw.io, desde a entrada da planilha até a geração automatizada dos gráficos. O modelo também se encontra no formato .pdf e .DRAWIO na pasta 04_docs dentro do projeto no GitHub.

Detalhamento dos Scripts

pasta 03_scripts/ armazena o script principal do app.R e inclui UI e Server

Tabela de funções utilizadas em app.R

Pacote	Função	Principais argumentos	Descrição de uso
base (R)	set.seed()	seed: um número inteiro usado para garantir reproduzibilidade da aleatoriedade	Torna os resultados aleatórios reproduzíveis.
base (R)	runif()	n: número de valores; min: valor mínimo; max: valor máximo (gera distribuição uniforme contínua)	Gera números aleatórios com distribuição uniforme.
base (R)	rnorm()	n: número de valores a gerar; mean: média da distribuição normal; sd: desvio padrão da distribuição normal	Gera números aleatórios com distribuição normal.
base (R)	cumsum()	x: vetor numérico no qual se deseja calcular a soma acumulada	Retorna a soma acumulada dos elementos de um vetor.
base (R)	sum()	x: vetor numérico; na.rm = TRUE para ignorar valores NA se necessário	Retorna a soma dos elementos de um vetor.
base (R)	paste0()	vetores de caracteres ou expressões para concatenar sem espaço entre elas	Concatena strings sem espaço entre elas.
wakefield	r_data_frame()	n: número de linhas da base; as demais colunas são definidas por funções auxiliares	Cria um data frame com n linhas e colunas simuladas usando funções auxiliares do pacote.
wakefield	r_sample_integer()	n: número de elementos a sortear; x: vetor de inteiros possíveis para o sorteio	Gera uma amostra de inteiros dos valores fornecidos.
dplyr	mutate()	coluna = expressão: cada nova coluna é criada ou substituída a partir de expressões envolvendo as colunas existentes	Adiciona ou modifica colunas.
dplyr	case_when()	condição ~ valor: cada linha avalia as condições fornecidas e aplica o valor correspondente; .default: valor usado quando nenhuma condição é satisfeita	Cria valores condicionais baseados em lógica tipo if/else .
dplyr	row_number()	retorna o número da linha	Retorna o número da linha dentro de cada grupo.
dplyr	group_by()	uma ou mais variáveis para agrupar os dados	Agrupa o data frame por uma ou mais variáveis para operações por grupo.
dplyr	arrange()	variáveis pelas quais ordenar; usar desc() para ordem decrescente	Ordena as linhas com base em variáveis.
dplyr	ungroup()	remove qualquer agrupamento anterior criado com group_by()	Remove agrupamentos anteriores.
dplyr	select()	nomes ou posições das colunas a manter e/ou renomear	Seleciona ou renomeia colunas específicas.
purrr	walk()	.x é o vetor ou lista de entrada, e .f é a função a ser aplicada a cada elemento	Aplica uma função a cada elemento de um vetor ou lista
readxl	read_excel()	path é o caminho do arquivo; sheet escolhe a aba; col_names define os nomes das colunas; range limita o intervalo; skip ignora linhas iniciais.	Importa dados de arquivos .xlsx de forma estruturada

pasta R/ armazena as funções modulares de cada gráfico separadamente

Tabela de funções utilizadas para construção dos gráficos			
Pacote	Função	Principais argumentos	Descrição de uso
dplyr	filter()	expressões lógicas para filtrar linhas	Filtre linhas da base com base em condições lógicas.
dplyr	summarise()	expressões para resumir colunas	Resume os dados agregando colunas para cálculos
dplyr	across()	cols: Seleção de colunas (everything(), where(is.numeric), etc.); .fns: Função ou lambda; .names: Formato do nome resultante.	Aplica funções a múltiplas colunas ao mesmo tempo.
tidyverse	pivot_wider()	names_from: Coluna que fornecerá os nomes das novas colunas; values_from: Coluna que fornecerá os valores; values_fill: Valor para preencher NAs; values_fn: Função de agregação caso haja múltiplos valores por grupo.	Transforma dados longos em formato largo. Cria colunas com base em uma variável.
forcats	as_factor()	Vetor a ser transformado em fator	Transforma colunas em fatores, úteis para eixos categóricos nos gráficos.
echarts4r	e_charts()	Coluna usada para o eixo x ou agrupamento principal.	Inicializa o gráfico e define o eixo x.
echarts4r	e_bar()	serie: Valores a serem plotados como barras; bind: Valores para rótulo; name: Nome da série; label, stack, itemStyle, emphasis: listas de estilo.	Adiciona uma barra vertical ao gráfico. bind define o rótulo; itemStyle configura cor, borda e estilo da barra.
echarts4r	e_line()	Igual ao e_bar(), mas para linha. lineStyle pode controlar espessura, opacidade, tipo (sólido, tracejado).	Adiciona uma linha ao gráfico. Pode ser usada apenas como rótulo visual (linha invisível com opacity = 0)
echarts4r	e_y_axis()	show: Booleano para exibir ou esconder o eixo y; pode receber outros estilos como name, min, max, splitLine, etc.	Edita o eixo y
echarts4r	e_tooltip()	trigger: "axis" ou "item", define quando o tooltip será exibido; pode incluir formatter, backgroundColor, etc.	Ativa o tooltip ao passar o cursor sobre o eixo x (mostra todas as séries do ponto)
echarts4r	e_theme()	name: Nome do tema: "auritus", "dark", "vintage", "infographic", "macarons", etc.	Aplica um tema visual ao gráfico: cor de fundo, fonte, paleta, grid

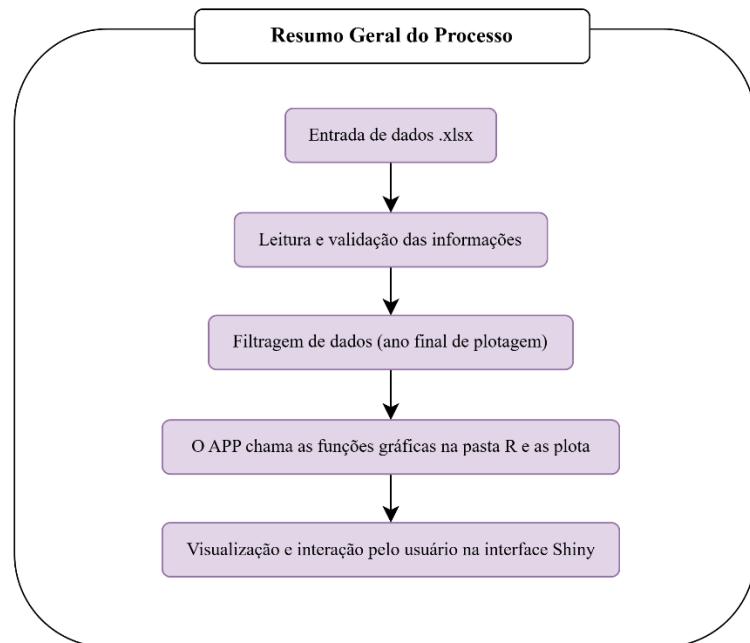


Figura 5: Fluxo de estruturação dos resultados. Fonte: Autora (2025).

Para acessar o aplicativo existem duas formas. Uma é escaneando o QR Code presente na Figura 6 ou clicando diretamente no URL: <https://flavialemos.shinyapps.io/aplicativo/>, e a outra é baixando o projeto R diretamente do GitHub, presente em: https://github.com/viallemos/20250516_relatorio_planejamento_florestal, seguindo a instrução da Figura 7. Assim é possível executar o aplicativo ao extrair o conteúdo da pasta .ZIP, abrir o projeto, clicar no script app.R presente na pasta 03_scripts, e clicar no botão destacado na Figura 8.



Figura 6: QR Code para abrir o aplicativo em HTML. Fonte: Resultados originais da pesquisa.

The screenshot shows the GitHub project page for '20250516_relatorio_planejamento_florestal'. The page includes navigation tabs like Código, Problemas, Ações, Projetos, Segurança, Percepções, and Configurações. Below the tabs, there's a search bar and a button to add files. The main area displays the repository structure with folders like 01_dados_brutos, 02_data_rds, 03_scripts, 04_docs, R, and renv. On the right side, there's a 'Code' section with options for Local, HTTPS, SSH, and GitHub CLI. The HTTPS link is highlighted with a red box. Below it, there are buttons for 'Clone', 'Clonar usando o URL da web.', 'Abrir com o GitHub Desktop', and 'Baixar ZIP', with the 'Baixar ZIP' button also highlighted with a red box.

Figura 7: Download do projeto via site GitHub através dos botões Código e Baixar ZIP. Fonte: Resultados originais da pesquisa.

```

1
2 # Load all
3
4 library(shiny)
5 library(bslib)
6 library(echarts4r)
7
8 funcoes <- list.files(here::here("R"), pattern = "\\.R$", full.names = TRUE)
9 purrr::walk(funcoes, .f = source) # puxa todas as funções da pasta R de uma vez
10
11 # Shiny app
12
13 # Ui
14
15 # Server

```

Figura 8: Botão “Run App” para abrir o aplicativo em HTML. Fonte: Resultados originais da pesquisa.

Uma interface inicial com as instruções de uso foi preparada, (Figura 9), e junto a ela, um exemplo de tabela com as informações que devem ser contidas na base de input do aplicativo para seu pleno funcionamento e geração de todos os gráficos.

Passo a passo para usar o app

1. Prepare seu arquivo Excel conforme o modelo abaixo.
2. Suba o arquivo .xlsx usando o botão “Browse...” da aba “Upload do arquivo”.
3. Verifique na pré-visualização se a sequência das informações estão corretas no seu documento de input.

Modelo aceito

- A planilha deve conter estas informações sequencialmente:

ID Talhão	Gênero	Idade atual (anos)	Ano de plantio	Idade de colheita (anos)	Ano de colheita	Distância de transporte até a fábrica (km)	Área (ha)	Volume (m³)	Modalidade
-----------	--------	--------------------	----------------	--------------------------	-----------------	--	-----------	-------------	------------

! Os nomes das colunas não precisam ser idênticos aos do exemplo, contanto que as informações correspondam.

Exemplo de uma fração de tabela:

Figura 9: Página 1 do aplicativo: Instruções de uso. Fonte: Resultados originais da pesquisa.

O uso do aplicativo acontece de forma direta, depois de carregar uma planilha padrão clicando no botão “Browse...”, destacado na Figura 10, os dados ficam disponíveis para pré-visualização para checagem de colunas, como visto na Figura 11. Após o carregamento os gráficos das próximas páginas do app se atualizam automaticamente e as informações plotadas variam a depender do ano escolhido pelo usuário como limite, permitindo uma análise rápida de vários cenários temporais, basta deslizar o botão *slider* para o ano limite que se pretende visualizar, (Figura 12).

INSTITUTO DE Florestas Relatório - Planejamento Florestal

Instruções Upload do arquivo Volume de Eucalipto (mil/m³) Faturamento (mi R\$) Idade de Colheita (anos)

Distância Média de Transporte (Km) Colheita Jovem (mil/m³)

Verifique se as informações contidas na tabela seguem esta ordem:

1. Talhão
2. Gênero
3. Idade atual
4. Ano de plantio
5. Idade de colheita
6. Ano de colheita
7. Distância de transporte até a fábrica
8. Área
9. Volume
10. Modalidade

Selecione seu arquivo (.xlsx):

Browse... No file selected

Figura 10: Página 2 do aplicativo: Upload do arquivo. Fonte: Resultados originais da pesquisa.

INSTITUTO DE Florestas Relatório - Planejamento Florestal

Instruções Upload do arquivo Volume de Eucalipto (mil/m³) Faturamento (mi R\$) Idade de Colheita (anos)

Distância Média de Transporte (Km) Colheita Jovem (mil/m³)

Y. volume
10. Modalidade

Selecionar seu arquivo (.xlsx):

Browse... dados_brutos.xlsx

Upload complete

talhao	genero	idade_atual	ano_plantio	idade_colheita	ano_colheita	dmt	area_ha	volume_m3	modalidade
183	Eucalyptus	12.00	2013.00	12.00	2025.00	92.24	52.03	20971.44	compra
t6	Eucalyptus	11.00	2014.00	11.00	2025.00	88.67	55.19	20604.86	compra
t36	Eucalyptus	12.00	2013.00	12.00	2025.00	54.12	44.67	18657.92	compra
t88	Eucalyptus	10.00	2015.00	10.00	2025.00	72.23	50.13	18117.28	compra
t59	Eucalyptus	11.00	2014.00	11.00	2025.00	58.73	51.11	17823.83	compra
t71	Eucalyptus	10.00	2015.00	10.00	2025.00	76.14	53.00	17406.22	compra
t18	Eucalyptus	10.00	2015.00	10.00	2025.00	97.87	51.40	17010.86	compra
t23	Eucalyptus	13.00	2012.00	13.00	2025.00	110.24	35.23	16806.26	compra
t61	Eucalyptus	11.00	2014.00	11.00	2025.00	74.21	42.50	15807.84	compra
t56	Eucalyptus	8.00	2017.00	8.00	2025.00	72.95	55.05	15685.40	compra

Figura 11: Página 2 do aplicativo: Pré visualização dos dados. Fonte: Resultados originais da pesquisa.



Figura 12: Página 5 do aplicativo: Idade de Colheita (anos) e destaque para slider. Fonte: Resultados originais da pesquisa.

Um exemplo de interpretação, visível no gráfico de colheita jovem, permite identificar volumes colhidos abaixo da idade de corte ideal, ao filtrar os dados até um certo ano, como visto na Figura 13, sugerindo, antecipações forçadas e potenciais erros no planejamento. A leitura integrada é feita relacionando gráficos de volume (Figura 14), idade de colheita, faturamento (Figura 15), e a distância média de transporte DMT (Figura 16).

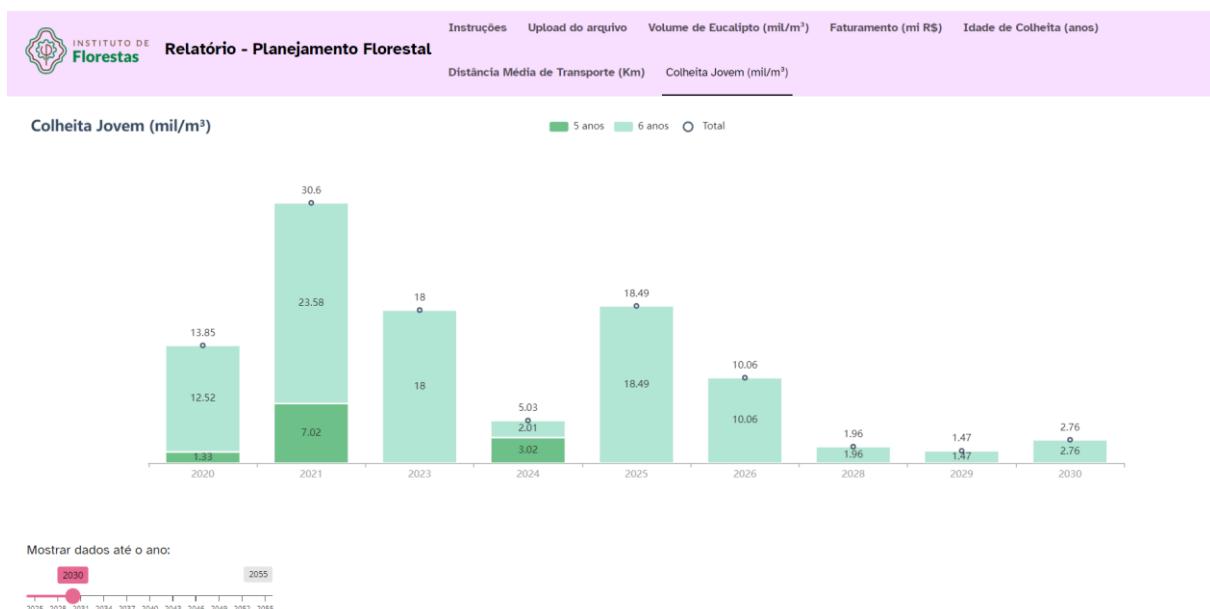


Figura 13: Página 7 do aplicativo: Colheita Jovem (mil m³). Fonte: Resultados originais da pesquisa.



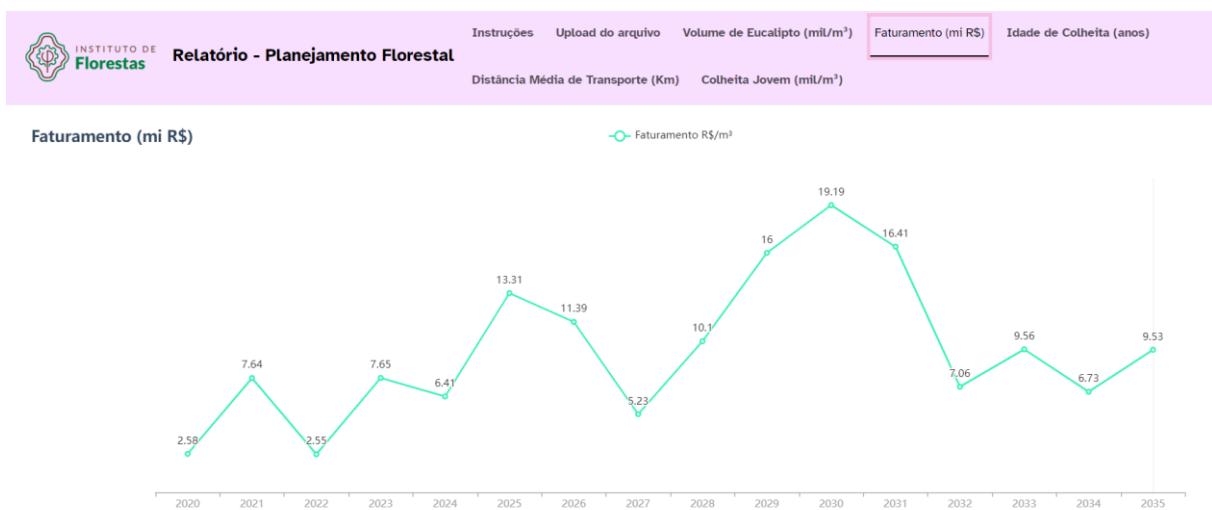
Volume de Eucalipto (mil/m³)

Base própria Compra Total

Mostrar dados até o ano:

2025 2035 2055

Figura 14: Página 3 do aplicativo: Volume de Eucalipto (mil m³) e destaque para uma classe. Fonte: Resultados originais da pesquisa.



Faturamento (mi R\$)

Faturamento R\$/m³

Mostrar dados até o ano:

2025 2035 2055

Figura 15: Página 4 do aplicativo: Faturamento (mi R\$). Fonte: Resultados originais da pesquisa.

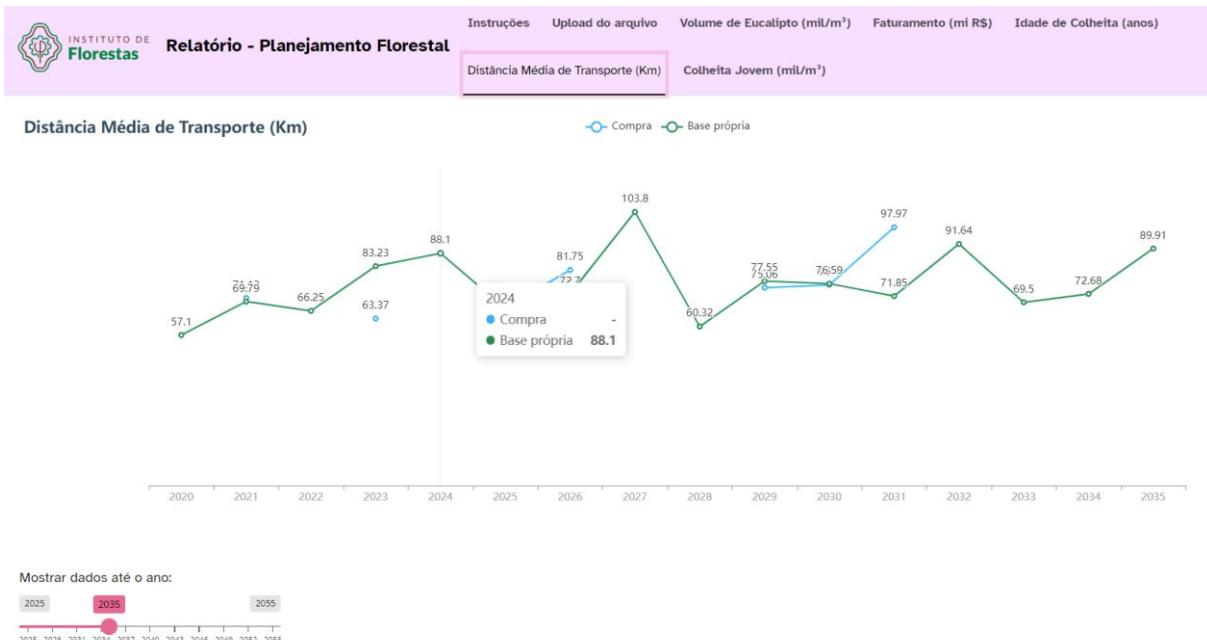


Figura 16: Página 6 do aplicativo: Distância Média de Transporte (km). Fonte: Resultados originais da pesquisa.

Comparado com as soluções da literatura, usando painéis construídos com *Visual Basic for Applications* (VBA), uma linguagem de programação desenvolvida pela Microsoft que utiliza recursos integrados para edição de macros, (Microsoft, 2025), e usando Power BI, que apresenta uma curva de aprendizado acentuada devido à complexidade de suas linguagens (*Data Analysis Expressions* (DAX) e Power Query) e limitações como a necessidade de licenciamento (CSS Learning Academy, 2024), a solução apresentada se destaca por utilizar ferramentas gratuitas, sendo independente de *software* proprietário, e ainda de fácil manutenção.

No entanto, alguns impasses existem. O uso de dados simulados não atrapalha a aplicação, bastando que o usuário siga o formato de entrada estabelecido. Para dados reais a aplicação exige uma estrutura de dados definida caso se queira adicionar novos gráficos e tratamentos de dados, sendo preciso um conhecimento mínimo de R para os ajustes ou só pra manutenção, além de que não tem, por hora, um banco de dados real integrado.

Por outro lado, a proposta é promissora, e possível de expandir. A plataforma funciona em ambiente online, e tem interface que ajuda a colaboração entre usuários e armazenamento via conta GitHub, mapeando alterações, adições e retiradas de materiais e linhas dos scripts como visualizado na Figura 17. Ademais, é útil para outras análises florestais, por exemplo, para indicadores de certificação, para monitoramento ambiental, ou até as projeções de estoque a longo prazo.

```

dados_filtrados <- dplyr::filter(
  dados,
  ano_colheita >= ano_min,
  ano_colheita <= input$ano_final
)
validate(
  @@ -142,7 +185,182 @@ server <- function(input, output) {
    graph_abastecimento(dados_filtrados)
    # graph_abastecimento(dados)
  })
+
+
+ output$graph_faturamento <- renderEcharts4r({
+   validate(
+     need(input$arquivo, "Envie um arquivo para visualizar o gráfico.")
+   ) # se nenhum arquivo for enviado o gráfico não é renderizado e envi
+   a a mensagem
})
  
```

Figura 17: Alterações de scripts no Github onde o grifo em vermelho simboliza a exclusão da linha do script e o grifo em verde simboliza a adição. Fonte: Resultados originais da pesquisa.

Somado a isso, este modelo conceitual poderá ser capaz de congregar ferramentas que sejam executadas automaticamente nos processos internos do planejamento florestal, fortalecendo uma maior integração, agilidade e também confiabilidade na análise de dados operacionais, além de sua ampla divulgação às partes interessadas.

5. CONCLUSÃO

Este estudo propôs e implementou um modelo conceitual para automação da análise de dados no planejamento florestal, utilizando ferramentas gratuitas, de código aberto e replicáveis para geração de um aplicativo que apresenta indicadores estratégicos de maneira gráfica e interativa. A arquitetura desenvolvida demonstrou ser eficaz na sistematização de fluxos operacionais, levando automação, padronização das análises e facilidade na visualização e uso dos dados, representando um avanço em direção à modernização das rotinas analíticas no setor, abrindo caminhos para aprimoramentos futuros conforme novas demandas e contextos forem incorporados.

A publicação do projeto em repositório público no GitHub favorece o uso colaborativo e a continuidade do desenvolvimento por outras equipes. Como possibilidades de trabalhos futuros, destacam-se a novas visualizações gráficas com a incorporação de dados econômicos, a integração com bases geoespaciais, a conexão com sistemas de inventário florestal e a implementação de novos filtros visuais, atendendo às necessidades que surgirem com a utilização da ferramenta.

6. REFERÊNCIAS BIBLIOGRÁFICAS

ANGELO, H. *et al.* Determinantes para um sistema de informação florestal. Floresta e Ambiente, Seropédica, v. 24, p. e0033, 2017. DOI: 10.1590/2179-8087.0033. Disponível em: <https://www.scielo.br/j/floram/a/q86ZtT5tbGrPf4pv7QCq49v/?lang=pt>. Acesso em: 22 abr. 2025.

ATKINS, J. W. *et al.* Open-Source tools in R for forestry and forest ecology. Forest Ecology and Management, v. 503, art. 119813, 2022. DOI: 10.1016/j.foreco.2021.119813. Disponível em: <https://doi.org/10.1016/j.foreco.2021.119813>. Acesso em: 17 jun. 2025.

BAUMER, B. S. R Markdown: integrating a reproducible analysis tool into introductory statistics. arXiv preprint, arXiv:1402.1894, 2014. Disponível em: <https://arxiv.org/abs/1402.1894>. Acesso em: 21 mai. 2025.

CARPENTER, B. Getting started with Stan. Disponível em: <https://bob-carpenter.github.io/stan-getting-started/stan-getting-started.html>. Acesso em: 28 jun. 2025.

CEPEA – Centro de Estudos Avançados em Economia Aplicada. Setor Florestal, nº 279, março de 2025. Piracicaba: ESALQ/USP, 2025. Disponível em: <https://www.cepea.esalq.usp.br/>. Acesso em: 16 jun. 2025.

CCS LEARNING ACADEMY. Top 10 Power BI Limitations You Should Know in 2025. 2024. Disponível em: <https://www.ccslearningacademy.com/top-power-bi-limitations/>. Acesso em: 26 jun. 2025.

CHACON, S.; STRAUB, B. Pro Git. 2. ed. Berkeley: Apress, 2014. Cap. 2: Git Basics. Disponível em: <https://git-scm.com/book/en/v2>. Acesso em: 21 jun. 2025.

CURSO-R. O pacote lubridate. Disponível em: <https://livro.curso-r.com/7-5-o-pacote-lubridate.html>. Acesso em: 26 jun. 2025.

CURSO-R. Os pacotes readxl e writexl. Disponível em: <https://livro.curso-r.com/5-3-readxl.html>. Acesso em: 26 jun. 2025.

DPLYR. Introduction to dplyr. Disponível em: <https://dplyr.tidyverse.org/articles/dplyr.html>. Acesso em: 26 jun. 2025.

FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS (FAO). Revisiting the concept of planning in relation to forests. FAO, 2003. Disponível em: <https://www.fao.org/4/w3210e/w3210e03.htm>. Acesso em: 20 jun. 2025.

FSC – FOREST STEWARDSHIP COUNCIL. Controlled Wood Standard – FSC Mix label percentage. Disponível em: <https://ca.fsc.org/ca-en/controlled-wood>. Acesso em: 16 maio. 2025.

IBÁ – INDÚSTRIA BRASILEIRA DE ÁRVORES. Relatório Anual 2024. São Paulo: IBÁ, 2024. Disponível em: <https://iba.org/datafiles/publicacoes/relatorios/relatorio-anual-iba2024.pdf>. Acesso em: 21 jun. 2025.

GIT-SCM. Git – Fast Version Control System. 2024. Disponível em: <https://git-scm.com/>. Acesso em: 2 fev. 2025.

GITHUB. GitHub Documentation. 2024. Disponível em: <https://docs.github.com/>. Acesso em: 2 fev. 2025.

KACZMAREK, I. J. Desenvolvimento de recursos computacionais derivados da programação de computadores para empresas do setor de florestas. 2021. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Florestal) – Universidade Federal do Paraná, Curitiba, 2021.

KLABIN S/A. Sobre a Klabin. 2025. Disponível em: <https://klabin.com.br/our-essence/about-klabin>. Acesso em: 2 fev. 2025.

LAI, J. *et al.* The use of R in forestry research. Journal of Plant Ecology, v. 16, n. 6, 2023. Disponível em: <https://doi.org/10.1093/jpe/rtad047>. Acesso em: 17 jun. 2025.

MICROSOFT. Getting started with VBA in Office. 2025. Disponível em: <https://learn.microsoft.com/pt-br/office/vba/library-reference/concepts/getting-started-with-vba-in-office>. Acesso em: 26 jun. 2025.

MINAS GERAIS. Instituto Estadual de Florestas. Manual do PSS e CAS: Procedimentos de Regularização Ambiental. Belo Horizonte: IEF, 2012. Disponível em: https://ief.mg.gov.br/documents/d/ief/manual_do_pss_e_cas-pdf. Acesso em: 10 jun. 2025.

OLIVEIRA FILHO, P. C. et al. Implementação de um sistema de informação geográfica para a gestão da empresa florestal. Floresta, Curitiba, v. 33, n. 1, p. 31–52, jan./abr. 2003. Disponível em: <https://doi.org/10.5380/rf.v33i1.2309>. Acesso em: 12 jun 2025.

PLOTLY. Bar plots in ggplot2. Disponível em: <https://plotly.com/ggplot2/bar-charts/>. Acesso em: 26 jun. 2025

PLOTLY. Creating and updating figures in R. Disponível em: <https://plotly.com/r/creating-and-updating-figures/>. Acesso em: 26 jun. 2025.

POSIT. Layout Guide. 2024b. Disponível em: <https://shiny.posit.co/r/articles/build/layout-guide/>. Acesso em: 20 jun. 2025.

POSIT. Lesson 2: Structure of a Shiny App. 2024a. Disponível em: <https://shiny.posit.co/r/getstarted/shiny-basics/lesson2/>. Acesso em: 20 jun. 2025.

POSIT. reticulate: Interface to Python. 2025. Disponível em: <https://rstudio.github.io/reticulate/>. Acesso em: 26 jun. 2025.

POSIT. RStudio IDE User Guide. 2023. Disponível em: <https://docs.posit.co/ide/user/2023.06.2/>. Acesso em: 21 jun. 2025.

RCPP CORE. Rcpp: Seamless R and C++ Integration. Disponível em: <https://rcpp.org/>. Acesso em: 26 jun. 2025.

R FOUNDATION. The R Project for Statistical Computing. 2024. Disponível em: <https://www.r-project.org/>. Acesso em: 2 fev. 2025.

R GRAPH GALLERY. ggiraph: Interactive graphics with ggplot2. Disponível em: <https://r-graph-gallery.com/package/ggiraph.html>. Acesso em: 28 jun. 2025.

R LIB. here: A package for file paths in R projects. Disponível em: <https://here.r-lib.org/articles/here.html>. Acesso em: 26 jun. 2025.

ROBINSON, E. Introduction to forcats. Disponível em: <https://forcats.tidyverse.org/articles/forcats.html>. Acesso em: 26 jun. 2025.

RSTUDIO. Dashboards com bslib. Disponível em:
<https://rstudio.github.io/bslib/articles/dashboards/index.html>. Acesso em: 26 jun. 2025.

RSTUDIO. R Markdown: Integrating R code with LaTeX. 2014. Disponível em:
https://rmarkdown.rstudio.com/articles_integration.html. Acesso em: 26 jun. 2025.

RSTUDIO. Using renv with RStudio. Disponível em:
<https://rstudio.github.io/renv/articles/renv.html>. Acesso em: 26 jun. 2025.

TIDYVERSE. purrr: Functional Programming Tools. Disponível em:
<https://purrr.tidyverse.org/reference/index.html>. Acesso em: 26 jun. 2025.

TIDYVERSE. tidyr: Tidy Messy Data. Disponível em:
<https://tidyr.tidyverse.org/reference/index.html>. Acesso em: 26 jun. 2025.

TRINKER, T. wakefield: Generate random data sets. Disponível em:
<https://github.com/trinker/wakefield>. Acesso em: 26 jun. 2025.