

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Kevin Klein	SP3096289
Luiz Fernando	SP3096301
Ruan de Souza	SP3069672
Pedro Dias	SP3099211

STUDY FLOW

São Paulo - SP - Brasil

2024

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Kevin Klein	SP3096289
Luiz Fernando	SP3096301
Ruan de Souza	SP3069672
Pedro Dias	SP3099211

STUDY FLOW

Documento final da aplicação.

Professor: JOHNATA SOUZA SANTICIOLI

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Tecnologia em Análise e Desenvolvimento de Sistemas

PI1A5 - Projeto Integrado I

São Paulo - SP - Brasil

2024

Resumo

O objetivo deste projeto é a implementação de um sistema que auxilia estudantes a se prepararem para um concurso público. Para isso, buscamos através das referências bibliográficas, teóricas e de mercado, como uma boa ferramenta de organização de estudos deve ser, incluindo funcionalidades essenciais e funcionalidades que nos diferem de concorrentes. Para atingirmos o objetivo do projeto, entendemos que a união de tecnologias de Inteligência Artificial e metodologia de organização de estudos ajudam os estudantes a se preparem melhor para as suas provas.

Palavras-chaves: Inteligência Artificial, Rotina de estudo, Concurso Público, Organização de Estudo.

Abstract

The objective of this project is to implement a system that helps students prepare for a public examination. To do this, we searched through bibliographical, theoretical and market references to see what a good study organization tool should look like, including essential features and features that differentiate us from competitors. In order to achieve the project's objective, we believe that the combination of Artificial Intelligence technologies and study organization methodology helps students to better prepare for their exams.

Key-words: Artificial Intelligence, Routine, Public Tender, Study Organization.

Lista de ilustrações

Figura 1 – QR Code do Blog:	20
Figura 2 – QRCode do Youtube	21
Figura 3 – Arquitetura de Sistema	25
Figura 4 – Modelo de classes do banco de dados	28
Figura 5 – QRCode do Deploy	47

Lista de tabelas

Tabela 1 – Métricas do projeto	24
Tabela 2 – Custos Fixos	44
Tabela 3 – Custos Variaveis	44

Lista de quadros

Quadro 1 – Atividades de Desenvolvimento	19
Quadro 2 – Atividades de gestão e planejamento	19
Quadro 3 – Análise de Riscos	21
Quadro 4 – Regras de Negócio	34
Quadro 5 – Requisitos Funcionais	35
Quadro 6 – Requisitos Não Funcionais	36

Lista de abreviaturas e siglas

API	<i>Application Programming Interface</i> - Interface de programação de aplicação - Citado em 26 , 27 , 44 , 47
COVID	<i>Corona Virus Disease</i> - Doença do Corona Virus - Citado em 11
CSS	<i>Cascading Stylesheet</i> - Folha de Estilo em cascata - Citado em 25 , 26
EAD	Educação a Distância - Citado em 11
HTML	<i>Hypertext Markup Language</i> - Linguagem de Marcação de Hipertexto - Citado em 25
HTTP	<i>Hypertext Transfer Protocol</i> - Protocolo de Transferência de Hipertexto - Citado em 27
IA	<i>Artificial Intelligence</i> - Inteligência Artificial - Citado em 11 , 15 , 16 , 17 , 22 , 25 , 26 , 27 , 30 , 36 , 40 , 46
ID	<i>Identity</i> - Identificador - Citado em 41 , 42
IFSP	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Citado em 18
JSON	<i>JavaScript Object Notation</i> - Notação de objetos JavaScript - Citado em 46
JWT	<i>JSON WEB Tokens</i> - Tokens JSON WEB - Citado em 41 , 42
LGPD	Lei Geral de Proteção de Dados - Citado em 40 , 41
MVP	<i>Minimum Viable Product</i> - Produto Mínimo Viável - Citado em 46
NLP	<i>Natural Language Processing</i> - Processamento de Linguagem Natural - Citado em 30
NPM	<i>Node Package Manager</i> - Gerenciador de Pacotes Node - Citado em 33
PDF	<i>Portable Document Format</i> - Formato de Documento Portável - Citado em 26
PoC	<i>Proof of Concept</i> - Prova de Conceito - Citado em 39 , 40 , 45 , 46
REST	<i>Representational State Transfer</i> - Transferência Representacional de Estado - Citado em 27
WEB	<i>World Wide Web</i> - Rede com Amplitude Mundial - Citado em 12 , 24
WIP	<i>Work in Progress</i> - Trabalho em progresso - Citado em 20
YARN	<i>Yet Another Resource Negotiator</i> - Mais um negociador de recursos - Citado em 33

Sumário

1	INTRODUÇÃO	11
1.1	Objetivo	12
1.2	Justificativa	12
1.3	Análise dos Concorrentes	12
1.3.1	QConcursos	13
1.3.2	Estuda Aqui	13
1.3.3	Aprovado	14
1.3.4	Easy Study	14
1.4	Studyflow x Concorrentes	15
2	REVISÃO DA LITERATURA	16
2.1	Dificuldade em organizar rotina para estudos	16
2.2	A importância da Inteligência Artificial no apoio dos estudos	16
3	GESTÃO DO PROJETO	18
3.1	Formação da equipe	18
3.1.1	Papéis	18
3.1.2	Organização das atividades	18
3.2	Gestão de tempo e desenvolvimento	19
3.2.1	Scrum	19
3.2.2	Kanban	20
3.3	Gestão de comunicação	20
3.4	Análise de riscos	21
4	DESENVOLVIMENTO DO PROJETO	24
4.1	Métricas	24
4.2	Arquitetura da solução	24
4.2.1	Front-end	25
4.2.2	Back-end	27
4.2.3	Banco de dados	27
4.2.4	Integrações	30
4.2.5	Versionamento de Código	30
4.2.6	Infraestrutura	31
4.2.7	Escalabilidade	32
4.2.8	Convenções e Padronização de Código	32
4.3	Análise de Requisitos	34

4.3.1	Regras de Negócio	34
4.3.2	Requisitos Funcionais	35
4.3.3	Requisitos Não Funcionais	36
4.4	Histórias de Usuário	37
4.5	Fases de Entrega	39
4.6	Segurança	40
4.6.1	Autenticação e Autorização	41
4.6.2	Políticas de Segurança	42
4.7	Viabilidade Financeira	43
4.7.1	Custos	44
4.7.2	Planos de Assinatura e Expectativa Financeira	44
4.8	Prova de Conceito (POC)	45
4.8.1	Mudanças	45
4.9	Mínimo Produto Vável (MVP)	45
4.9.1	Mudanças na Infraestrutura	46
5	CONSIDERAÇÕES FINAIS	48
	REFERÊNCIAS	49

1 Introdução

No Brasil, temos um importante instrumento de ascensão social através dos estudos, chamado Concurso Público. Muitas pessoas se dedicam para essa esse tipo de prova, pois através dela pode-se ter melhores ganhos e estabilidade financeira, além de depender do concurso um emprego sem demissão e plano de carreira atrativo. Com esse cenário, milhares de brasileiros tentam todos os anos a prova para esses concursos públicos para diferentes cargos e diversas esferas do poder público, mas com toda essa gama de provas e quantidade de concorrentes, como se preparar da melhor forma?

Segundo [Bastos \(2019\)](#), a busca do público por concursos públicos vem aumentando em 40% ao ano. Isso deve aos altos investimentos que o governo faz ao setor, só no ano de 2023 foram anunciadas mais de 6 mil vagas, com um investimento total de R\$ 735 milhões [Sant'Ana \(2023\)](#). O tamanho de mercado e investimento, fez que surgissem empresas e plataformas que buscassem atender tamanha demanda de mercado, mas ainda assim não sendo suficientes frente a quantidade de pessoas interessadas em ingressar no serviço público. Com a Pandemia de *Corona Virus Diase* (COVID) iniciada em 2020, as plataformas tiveram que se adaptar as limitações da época e investir no modelo de [Educação a Distância \(EAD\)](#).

E o modelo [EAD](#) para ([NETTO; SOUZA, 2019](#)), têm sido de grande importância, pois é uma importante ferramenta de democratização ao conhecimento, se tratando principalmente de conteúdos preparatórios para concursos públicos. Com rotinas muitas vezes divididas entre trabalho e estudo, o estudante de concurso público vem adotando este modelo de estudo, pois possibilita maior flexibilidade de horários e menor rigidez em sua rotina. ([IAHN, 2008](#))

Apesar do modelo [EAD](#) ter ganhado força, ainda assim não é suficiente para suprir todas as dificuldades que os concurseiros enfrentam na jornada de estudos. Para [Salgado \(2016\)](#), entre as principais dificuldades dos concurseiros, estão o grande volume de matérias a se estudar e como organizar os seus estudos.

Pensando nessas dificuldades e no cenário de oportunidades, a plataforma Studyflow surge com a proposta de organizar a rotina de estudos do usuário, organizando e distribuindo as matérias a serem estudados a partir do edital do concurso desejado, utilizando a tecnologia de *Artificial Intelligence* (IA) na organização e rotina de estudos do concurseiro.

1.1 Objetivo

Este projeto tem como objetivo ajudar os estudantes de concurso público a se organizarem com as matérias e rotina de estudos para um concurso público com o auxílio de tecnologias de Inteligência Artificial.

A plataforma em ambiente [World Wide Web \(WEB\)](#) permitirá que os estudantes enviem o edital em que irão prestar a prova ou escrevam os conteúdos desse edital e a partir dessas informações a plataforma gere uma rotina de estudos semanal e personalizada, ajudando na preparação do usuário para a sua prova.

Como objetivos específicos da plataforma, foram definidos os seguintes:

- Aprovação de 50% dos usuários nos concursos;
- Conclusão de no mínimo 75% da rotina gerada semanalmente; e
- 30% dos usuários utilizando a ferramenta por 4 semanas seguidas.

Com os indicadores e objetivos descritos acima, podemos começar quantificar e analisar o impacto da ferramenta nas horas de estudo dos nossos usuários.

1.2 Justificativa

O projeto surge com o intuito de facilitação e otimização de tempo para os concurren-
seiros, com a utilização de recursos de inteligência artificial, métodos de estudo compro-
vados pela ciência e a criatividade, espera-se que a jornada e o resultado dos estudantes
seja mais fácil até o seu objetivo final que é a aprovação de um concurso.

A importância da aplicação vai além de sua praticidade, o seu valor agregado para o usuário é uma nova forma de utilizar a combinação entre estudo e tecnologia, que facilitará a concretização da jornada da aprovação em um concurso público.

1.3 Análise dos Concorrentes

A análise de concorrentes oferece uma análise exploratória de mercado, para enten-
der como os concorrentes e o mercado oferece de soluções quando falamos de plataformas
de gestão de estudos para concursos públicos. Buscamos trazer os principais concorrentes,
fazendo um comparativo das suas funcionalidades, tamanho de mercado e o que podemos
oferecer.

1.3.1 QConcursos

A plataforma do QConcursos é uma das principais plataformas em ensino e preparação para concursos públicos. A plataforma abrange desde dos concursos de nível municipal, até os concursos de escala nacional.

O QConcursos é ainda uma das maiores plataformas de concurso do Brasil, segundo números divulgado pela organização, eles possuem 28 Milhões de concurseiros, número esse que mostra a relevância da plataforma e de um potencial tamanho de mercado.

Entre suas principais funcionalidades estão o consumo de vídeoaulas, uma comunidade interativa para resolução de questões, mentorias individuais e um plano de estudo personalizado para o estudante. O QConcursos não é uma plataforma que foca mais no conteúdo e ensino das matérias para as provas do que na organização de rotina dos estudantes.

1.3.2 Estuda Aqui

Estuda Aqui é um aplicativo *web* e *mobile* que ajuda os alunos a organizar e otimizar seu aprendizado. O aplicativo oferece diversos recursos para ajudar os alunos a se organizarem e aproveitarem ao máximo o tempo de estudo.

Em planejamento, o aplicativo ajuda o aluno a criar um plano de estudos personalizado com base em diversos fatores, incluindo o exame que pretende fazer, o nível de conhecimento de cada matéria e o tempo disponível para estudar. O aplicativo também fornece recomendações de materiais de estudo e conteúdos relacionados.

Em controle de tempo, O Estuda Aqui permite que o aluno registre o tempo gasto em cada matéria e atividade. O aplicativo também fornece estatísticas detalhadas sobre o desempenho dos alunos ao longo do tempo, ajudando você a identificar pontos fortes e fracos para otimizar o aprendizado.

Oferece ainda um sistema de correção de distância baseado no esquecimento de curvas para ajudar os alunos a reter por mais tempo o que aprenderam. O aplicativo agenda revisões automaticamente com base na importância e complexidade do conteúdo e envia lembretes para que os alunos não se esqueçam de revisar.

E por fim, oferece uma comunidade online onde os alunos podem compartilhar informações e experiências. A comunidade também oferece grupos de aprendizagem online para diversas especialidades e oportunidades de colaboração com professores e especialistas.

1.3.3 Aprovado

O Aprovado é uma plataforma completa de estudo para concursos públicos, destinada a ajudá-lo a obter aprovação com máxima eficiência e tranquilidade. Mais do que um aplicativo, o Aprovado é seu parceiro de jornada, fornecendo as ferramentas e o suporte necessários para atingir seus objetivos.

Entre suas principais características estão:

- Plano de estudos inteligente: o Aprovado cria um plano de estudos personalizado e adequado à apresentação do concurso que você deseja estudar, ao seu nível de conhecimento e ao tempo disponível;
- Conteúdo Estratégico e Completo: vídeoaulas, materiais de estudo e simulações de especialistas renomados são totalmente focados nos temas e conteúdos que você precisa para competir;
- Acompanhamento detalhado: monitore seu progresso e identifique pontos fortes e fracos para otimizar sua pesquisa com relatórios e estatísticas abrangentes;
- Ensino à distância: o Aprovado utiliza métodos comprovados de memorização para ajudá-lo a reter o que aprendeu a longo prazo. Simulações realistas: pratique com simulações que imitam o formato real do exame para que você possa estar melhor preparado no dia da prova;
- Comunidade envolvente: Forme grupos de estudo para se conectar com outros candidatos, compartilhar experiências, ficar motivado e inspirado e ter a oportunidade de colaborar com professores e especialistas em sua área de especialização.

1.3.4 Easy Study

O Easy Study é um app para os estudos, trazendo diversas funcionalidades para ajudar os estudantes a organizar e melhorar seus métodos de aprendizagem. O destaque do aplicativo está em sua abordagem personalizada e flexível, que se adapta às necessidades individuais de cada usuário.

Para o estudante, a plataforma elabora planos personalizados com base em vários aspectos, incluindo o ritmo de aprendizado do usuário, suas metas e o tempo disponível. Além disso, o app sugere materiais de estudo e conteúdos relevantes.

Uma das suas principais funcionalidades é a variedade de simulados dos concursos para que os usuários testem seus conhecimentos e se acostumem com o formato das provas reais. Com correção automática, o aplicativo oferece um feedback detalhado sobre o desempenho do usuário.

1.4 Studyflow x Concorrentes

Em frente aos nossos quatro principais concorrentes, Easy Study, QConcursos, Aprovado e Estuda Aqui, os nossos diferenciais como plataforma são:

- Utilização de Inteligência Artificial para a organização de rotina de estudos;
- Utilização do edital do concurso para gerar o cronograma de estudos de forma automática;

O que a nossa plataforma não faz, mas que as outras plataformas fazem:

- Banco de dados de questões dos concursos;
- Videoaula ou materiais que ensinam sobre algum tema dos concursos;
- Mentoria feita por humano de como preparar a rotina para estudo; e
- Comunidade para troca de informações e dicas de estudo.

Os concorrentes de mercado possuem funcionalidades abrangentes, mas analisando e entendendo a funcionalidade de cada um, nenhum deles foca de forma profunda na organização da rotina do estudante ou organiza o cronograma de estudos (deixa a cargo do estudante ou oferece uma mentoria). Entendemos que o ponto forte e de diferencial da plataforma é a praticidade de organizar uma rotina de estudos de forma automática, utilizando a [IA](#) como ferramenta de apoio, ajudando a economizar tempo com essa tarefa.

2 Revisão da Literatura

A seção de revisão da literatura tem como objetivo apresentar pesquisas, artigos, livros ou afins já realizados por outros autores, que embasem as problematizações utilizadas como motivo para o desenvolvimento da aplicação StudyFlow. Isto é, serão abordadas fontes de informação que comprovam a utilidade e pertinência da aplicação.

2.1 Dificuldade em organizar rotina para estudos

Com o crescimento exponencial da internet, cada vez mais informações são disponibilizadas ao público. No entanto, para aqueles que buscam conteúdo específico para estudar para concursos públicos, muitos dos resultados apresentados em uma busca regular em mecanismos de pesquisa podem ser irrelevantes ou incompletos.

Além da dificuldade em encontrar o conteúdo relevante para os concursos, os estudantes enfrentam desafios adicionais ao tentar organizar todas as informações. Esta é uma das maiores barreiras para aqueles que ainda não adotaram nenhum método ou rotina de estudos. (SALGADO, 2016) destaca: "Esse é o primeiro impacto, mesmo. É assustador. Eu senti isso na pele quando comecei a minha preparação." sobre o volume de conteúdo a ser estudado para concursos. "A solução é organizar o estudo, planejar a rotina diária para ter o momento certo de estudar e distribuir as matérias ao longo da semana."

Para superar as dificuldades citadas anteriormente, os estudantes podem recorrer a métodos de estudo, que ajudam a mitigar os problemas de organização e desempenho (SANTOS, 2015). Os benefícios para os estudantes que organizam sua rotina de estudos são diversos, entre eles o melhor desempenho acadêmico e satisfação nas atividades estudantis (LEITE, 2003). Dentre os métodos de estudo que os estudantes podem utilizar, eles são classificados em técnicas de memorização, concentração, organização de tempo, organização de matérias, entre outros.

2.2 A importância da Inteligência Artificial no apoio dos estudos

A IA está revolucionando diversos setores da sociedade, e a educação não é exceção. Nos últimos anos, ferramentas e plataformas impulsionadas por IA vêm surgindo com o objetivo de auxiliar os alunos em sua jornada de aprendizado, tornando-a mais personalizada, eficiente e eficaz.

Em destaque, os principais benefícios da IA na educação é a sua capacidade de personalizar o ensino de acordo com as necessidades individuais de cada aluno. Através de

algoritmos de aprendizado de máquina, os sistemas de [IA](#) podem analisar o desempenho, estilo de aprendizado e ritmo de cada estudante, adaptando o conteúdo, as atividades e os métodos de ensino de forma otimizada ([SHEMSHACK; SPECTOR, 2020](#)).

A utilização de ferramentas impulsionadas por [IA](#) não é novidade no mercado, mas a partir de 2022 houve um boom nesse mercado com a chegada do ChatGPT, criado pela OpenAI. Para fins estudantis, os *chatbots* são as ferramentas amplamente adotadas, por sua facilidade de desenvoltura e compreensão pelo usuário. Os Chatbots podem responder a dúvidas sobre a matéria, sistemas de tutoria oferecem exercícios personalizados e plataformas de aprendizado adaptativo sugerem conteúdos relevantes para cada estudante ([SILVA, 2023](#)). Além disso, ferramentas de reconhecimento de fala e texto podem auxiliar alunos com deficiências e softwares de tradução podem facilitar o aprendizado em diferentes idiomas.

3 GESTÃO DO PROJETO

Nesta seção, serão apresentados os métodos escolhidos para a gestão do projeto e da equipe, com o objetivo de assegurar a melhor utilização possível do tempo, orçamento e recursos voltados para o projeto, para que esse possa ser concluído dentro do prazo estabelecido. Também serão levantados alguns riscos possíveis, a fim de que com o conhecimento dessas possibilidades, medidas possam ser tomadas para evitá-los.

3.1 Formação da equipe

A equipe foi formalizada durante as aulas da disciplina, porém já havia sido definida posteriormente. Todos os integrantes são alunos do curso de Tecnologia em Análise e Desenvolvimento de Sistemas, [IFSP](#), campus São Paulo. A equipe se baseia nos conhecimentos de cada um de seus membros com o objetivo de preencher as necessidades do projeto. Os participantes da equipe Noz são:

- **Kevin Klein**
- **Luiz Fernando Cavalcante de Faria**
- **Pedro Felipe da Silva Dias**
- **Ruan de Souza Cardoso Brito**

3.1.1 Papéis

Os papéis foram definidos através das habilidades dos membros, para que cada um pudesse atuar de maneira segura com seus conhecimentos, dessa maneira a organização e a fluidez do projeto são beneficiadas. As atividades são planejadas para que todos sejam responsáveis por alguma parte específica do projeto, podendo receber ajuda dos outros participantes caso seja necessário.

3.1.2 Organização das atividades

Dentro das atividades do projeto, a divisão está descrita no [Quadro 1](#).

Além das atividades de desenvolvimento, as tarefas de gestão e planejamento foram divididas conforme o [Quadro 2](#)

Quadro 1 – Atividades de Desenvolvimento

Atividades	Kevin	Luiz	Pedro	Ruan
Front-end	X			X
Back-end	X		X	
Dados		X		
UX/UI		X		
Documentação		X		X
Inteligência Artificial	X		X	

Fonte: Os autores.

Quadro 2 – Atividades de gestão e planejamento

Atividades	Kevin	Luiz	Pedro	Ruan
SVN	X			
LaTeX		X		
Blog				X
Youtube				X
Contato Orientador			X	
Kanban	X			
Apresentações		X		

Fonte: Os autores.

As atividades citadas nos quadros 1 e 2 podem ser realizadas por outros membros do grupo, caso algum integrante não possa realizar atividade que foi atribuída.

3.2 Gestão de tempo e desenvolvimento

A equipe decidiu aderir à utilização do Scrum como *framework* de gerenciamento, a fim de melhorar a organização durante o desenvolvimento do projeto. Ele foi escolhido pela sua eficiência e ampla utilização por diversas empresas no mercado, além de ser conhecido pelos integrantes do grupo, o que facilita sua implementação. Além disso também optamos pela utilização do Kanban, para garantir a eficiência na realização das tarefas e o cumprimento dos prazos.

3.2.1 Scrum

O *Scrum* é uma metodologia de desenvolvimento ágil amplamente empregada para lidar com a complexidade na criação de produtos. Este método valoriza a colaboração, a autonomia da equipe e a entrega progressiva e iterativa. Composto por uma série de práticas, papéis e artefatos, o Scrum promove a eficácia e a qualidade do trabalho realizado, impulsionando a entrega de valor de forma consistente ao longo do tempo.

3.2.2 Kanban

O Kanban é uma metodologia de gestão visual que teve origem no Japão e ganhou popularidade em diversos setores, como desenvolvimento de software, manufatura e serviços. O termo "Kanban" significa "sinal visual" em japonês, e essa abordagem se baseia na utilização de cartões ou post-its para representar unidades de trabalho e visualizar o fluxo do processo. Essa metodologia visa proporcionar transparência sobre o trabalho em andamento e controlar o *Work in Progress* (WIP) para otimizar a eficiência do sistema.

3.3 Gestão de comunicação

A comunicação é parte essencial para que tudo corra bem no projeto. Foram utilizados alguns meios para realizar esse diálogo entre a equipe.

Os meios de comunicação utilizados internamente foram o *Whatsapp*, aplicativo de mensagens instantâneas e chamadas de voz, foi usado para troca de mensagens durante as semanas, a fim de proporcionar agilidade e facilidade na comunicação, e o *Discord*, que é uma aplicação voltada para a comunicação, principalmente de grupos e comunidades, foi usado para as reuniões realizadas semanalmente.

Para a comunicação com o público foi criado um *blog*, na plataforma Blogger, onde são compartilhadas as atualizações semanais e informações relevantes sobre o projeto. O blog pode ser acessada em <<https://equipenoz.blogspot.com>> ou através do QR Code na figura 1.

Figura 1 – QR Code do Blog:



Fonte: Os autores.

Além do *blog*, é possível acompanhar o desenvolvimento do projeto e das apresentações pelos vídeos presentes no canal do Youtube dedicado ao projeto. Os vídeos podem ser acessados pelo link <<https://youtube.com/@noz-ifsp?si=aH22hYmd9B4Nj2l9>> ou escaneando o QR Code da figura 2.

Figura 2 – QRCode do Youtube



Fonte: Os autores.

O canal do YouTube também serve como uma linha do tempo para o projeto, sendo possível acompanhar as mudanças feitas na arquitetura, requisitos e demais atividades do escopo do projeto.

3.4 Análise de riscos

Em todo projeto, riscos de todos os tipos e magnitudes podem ocorrer, impactando de forma significativa o andamento do projeto. Para a construção do Studyflow, foram mapeados os riscos, o nível de impacto e a resposta aos mesmos, todos eles descritos no Quadro 3.

Quadro 3 – Análise de Riscos

Risco	Nível de Impacto	Resposta
Desistência pessoal	Alto	Aceitar
Problemas de Saúde	Alto	Aceitar
Conflitos Interpessoais	Médio	Mitigar
Comprometimento com outras tarefas	Médio	Mitigar
Mudança de Requisitos	Alto	Eliminar
Falhas de comunicação	Alto	Eliminar
Falta de conhecimento técnico	Médio	Mitigar
Problemas com conexão de rede	Baixo	Aceitar
Problemas com Hardware	Baixo	Aceitar
Escopo mal definido	Alto	Mitigar
Desempenho insatisfatório	Médio	Mitigar
Falhas de segurança	Médio	Mitigar
Falha em tecnologias externas	Médio	Aceitar
Problemas com o modelo de IA	Alto	Eliminar

Fonte: Os autores.

Os possíveis riscos que podem acontecer no projeto são:

- **Desistência pessoal:** Membros da equipe abandonam o projeto, causando lacunas na expertise e sobrecarregando os membros restantes.
- **Problemas de saúde:** Membros da equipe enfrentam problemas de saúde que afetam sua capacidade de contribuir para o projeto.
- **Conflitos interpessoais:** Desentendimentos ou tensões entre membros da equipe prejudicam a colaboração e a eficiência do projeto.
- **Comprometimento com outras tarefas:** Membros da equipe têm prioridades divididas entre várias tarefas ou projetos, resultando em atrasos ou falta de dedicação ao projeto em questão.
- **Mudança de requisitos:** Alterações nos requisitos do projeto após o início do desenvolvimento, levando a retrabalho e atrasos.
- **Falhas na comunicação:** Comunicação inadequada entre membros da equipe, clientes ou partes interessadas, levando a mal-entendidos e erros.
- **Falta de conhecimento técnico:** Membros da equipe não possuem as habilidades ou conhecimentos necessários para concluir com sucesso determinadas tarefas ou aspectos do projeto.
- **Problemas com conexão de rede:** Problemas com a conexão de rede afetam a colaboração remota ou o acesso a recursos necessários para o projeto.
- **Falhas de hardware:** *Hardware* essencial para o projeto falha, causando interrupções no desenvolvimento ou perda de dados.
- **Escopo mal definido:** Requisitos do projeto não estão claramente definidos desde o início, levando a confusão e revisões frequentes.
- **Desempenho insatisfatório:** O produto final não atende às expectativas de desempenho dos usuários, levando à insatisfação e possível rejeição.
- **Falha de segurança:** Vulnerabilidades de segurança no sistema comprometem a integridade ou a confidencialidade dos dados, resultando em riscos para os usuários e para a empresa.
- **Falha em tecnologias externas:** Dependência de tecnologias externas que podem falhar ou não atender às expectativas, afetando o desenvolvimento ou o funcionamento da aplicação.
- **Problema no treinamento da IA:** Dificuldades no treinamento de sistemas de inteligência artificial para alcançar os resultados desejados, resultando em desempenho inadequado ou inexato.

Ao longo do desenvolvimento do projeto, o grupo busca manter ações para mitigar todos os possíveis riscos no desenvolvimento do projeto. Como medida, adotou-se a reunião semanal, não sendo discutido apenas aspectos técnicos do projeto, buscando entender como cada membro da equipe sente-se em relação ao projeto e se estava confortável com as atribuições de tarefa.

4 Desenvolvimento do Projeto

Este tópico apresenta informações do produto, desenvolvimento do *software*, tempo para execução, ferramentas utilizadas, viabilidade e riscos.

4.1 Métricas

Na tabela 1, definimos métricas que acompanhamos ao longo do desenvolvimento do software.

Tabela 1 – Métricas do projeto

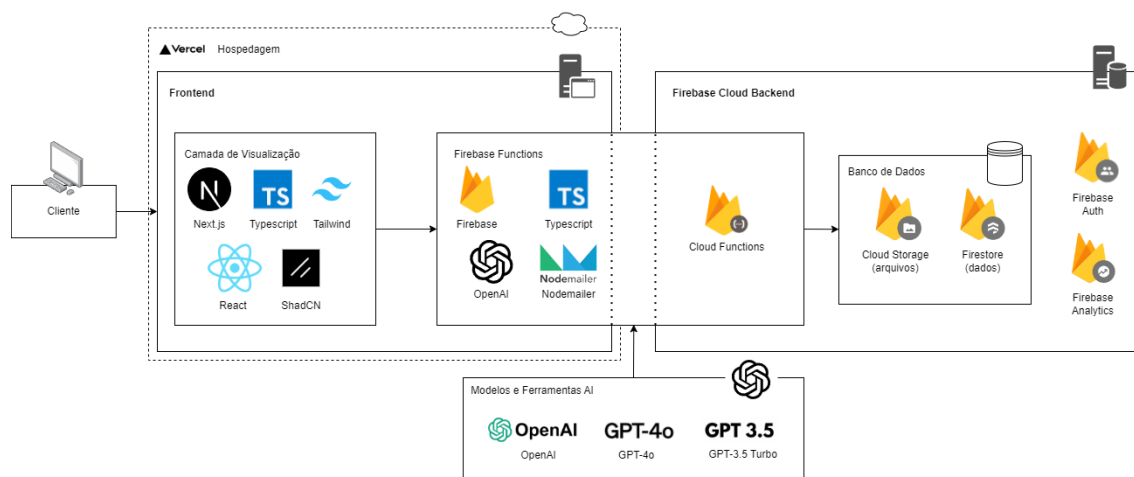
Itens	Progresso do Projeto						
	12/4	19/4	26/4	03/5	10/5	17/5	26/7
Reuniões	1	2	3	4	5	6	9
Posts de Blog	5	5	6	7	7	7	7
Vídeos	1	2	2	2	3	3	3
Requisitos	12	12	19	19	19	19	19
Entidades de BD	0	0	6	6	6	6	6
Commits	3	4	6	9	23	25	30

Ao longo das nossas revisões do projeto, como descrito na tabela 1, a aplicação foi finalizada com 12 requisitos, mas não foram 12 requisitos que foram discutidos ao longo do planejamento. Ao fim do projeto passamos por 19 requisitos, removendo e revisitando os que tinham sido definidos anteriormente.

4.2 Arquitetura da solução

O projeto [WEB](#) é dividido em duas camadas principais, sendo elas o *front-end*, responsável pela estilização da plataforma e interação com o usuário, e o *back-end*, responsável pela aplicação das regras de negócio, gestão das informações em um banco de dados, e pela lógica de execução em si da plataforma.

Figura 3 – Arquitetura de Sistema



Fonte: Os autores.

A arquitetura explicitada na figura 3, foi pensada de maneira em as tecnologias utilizadas não consumissem tanto tempo no desenvolvimento e se integrassem de forma natural, não precisando de diversas dependências ou criação de código de suporte para as tecnologias funcionassem entre si.

A principal decisão dessa arquitetura foi abstrair o *back-end* em uma tecnologia que necessita de pouco desenvolvimento, nos permitindo focar no desenvolvimento e integração da IA na aplicação.

4.2.1 Front-end

O *Front-end* é a interface da aplicação, construída com Next.js, um *framework* React que oferece otimizações de desempenho e server-side rendering para uma experiência de usuário mais rápida e fluida. No desenvolvimento da interface, são empregadas as seguintes tecnologias:

- **Tailwind CSS**

Tailwind CSS é uma ferramenta utilizada para a estilização da aplicação. Ele adota uma abordagem de *utility-first*, o que significa que as classes *Cascading Stylesheet* (CSS) são utilizadas diretamente no *Hypertext Markup Language* (HTML) para estilizar os elementos. Isso proporciona uma experiência de desenvolvimento mais rápida e consistente, além de facilitar a manutenção do código.

- **ShadCN**

ShadCN é uma coleção de componentes prontos que podem ser importados e customizados dentro do código. Esses componentes são escritos em *Typescript* e Tailwind

[CSS](#). Ele não é considerado uma biblioteca, já que é uma extensão do Radix, outra biblioteca de estilização para Javascript.

- **Moment**

Moment.js é uma biblioteca popular para manipulação de datas e horas em Javascript. Ela oferece uma ampla gama de funcionalidades para formatação, análise e manipulação de datas, tornando mais fácil trabalhar com informações temporais na aplicação.

- **Nodemailer**

Nodemailer é uma biblioteca utilizada para enviar e-mails através de Node.js. Ela oferece uma interface simples e flexível para o envio de *e-mails*, permitindo configurar facilmente o servidor de e-mail, criar *templates* personalizados e enviar mensagens de forma assíncrona.

- **Framer Motion**

Framer Motion é uma biblioteca de animações para React que facilita a criação de animações fluidas e responsivas em componentes da interface. Ela oferece uma [Application Programming Interface \(API\)](#) declarativa e intuitiva para definir animações de entrada, saída e transição, além de suportar gestos e interações do usuário.

- **PDF Viewer**

PDF Viewer é uma biblioteca Javascript projetada especificamente para a leitura de arquivos em formato [PDF](#) enviados pelos usuários, dentro do NodeJS. Com uma série de ferramentas avançadas, oferece uma experiência de visualização personalizada e intuitiva desses documentos.

- **Open AI**

A biblioteca OpenAI permite que os desenvolvedores usem os modelos de [IA](#) generativos de texto desenvolvidos pela OpenAI, como o GPT 3.5, para criar recursos e aplicativos com tecnologia de [IA](#).

- **Bibliotecas do Firebase**

Dentro do *Front-end*, são utilizadas diversas bibliotecas do Firebase para interação com o Back-end e execução de funcionalidades como autenticação, armazenamento de dados e comunicação em tempo real. Algumas das bibliotecas comumente utilizadas incluem:

- *firebase*
- *firebase-admin*
- *firebase-functions*

– *firebase-tools*

Essas bibliotecas fornecem uma integração simplificada entre o Front-end e o Back-end, permitindo o desenvolvimento de uma aplicação robusta e interativa.

4.2.2 Back-end

Para o Back-end, é utilizado Firebase, que fornece serviços de banco de dados, armazenamento, autenticação e hospedagem, entre outros, de forma simplificada. O Firebase permite uma configuração rápida e fácil, facilitando o desenvolvimento e a viabilização do projeto.

Pensando em maximizar o principal recurso do grupo, o tempo, discutimos uma forma de conseguir com o tempo disponível como desenvolver uma aplicação que utilizasse [IA](#) (que não é uma das nossas especialidades) e ainda lidar com as outras *stacks* necessárias para o projeto. O Firebase se apresentou como a melhor solução, por ser a mais fácil de ser apreendida e deespreende menos tempo para desenvolver uma aplicação, além de futuramente ajudar na escalabilidade do projeto.

Para construir o Studyflow, no Firebase as funcionalidades que estão sendo utilizadas:

- **Firebase Auth**

Para autenticação de usuários, permitindo login com *e-mail*, redes sociais, entre outros métodos.

- **Firebase Firestore**

Para armazenamento e gerenciamento de dados em tempo real, oferecendo um banco de dados NoSQL escalável e altamente disponível.

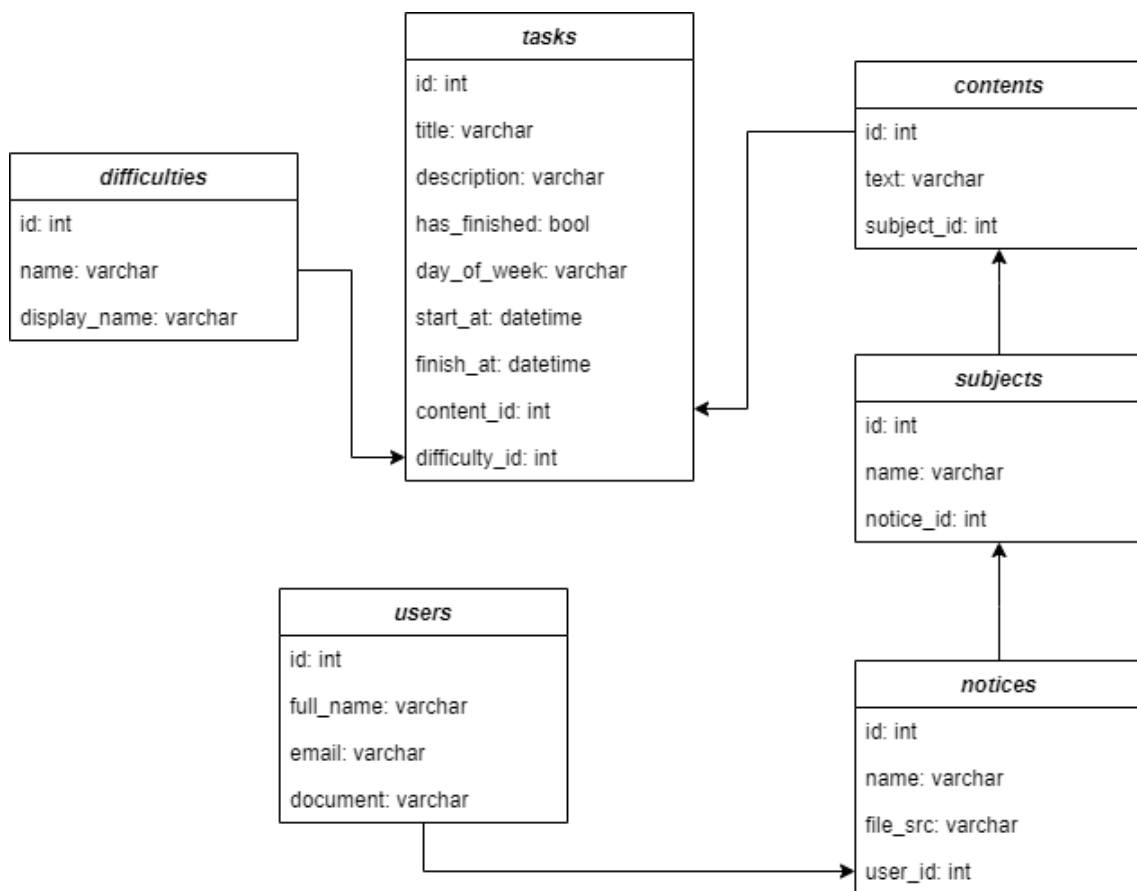
- **Firebase Storage**

Para armazenamento de arquivos, como imagens e vídeos, diretamente na infraestrutura do Firebase.

4.2.3 Banco de dados

O banco de dados da aplicação está contido nos serviços oferecidos pelo Firestore, que é um banco de dados NoSQL escalável e altamente disponível. A comunicação entre as camadas, [API](#) externas e com o cliente são realizadas através do Protocolo [Hypertext Transfer Protocol](#) (HTTP) e chamadas [Representational State Transfer](#) (REST).

Figura 4 – Modelo de classes do banco de dados



Fonte: Os autores.

A figura 4 mostra a modelagem do banco de dados atual, descrevendo todas as tabelas e colunas, assim como os tipos de dados de cada coluna. Escolhemos seguir com o banco NoSQL do Firestore pela flexibilidade no armazenamento das informações, especialmente para suportar os dados que são consumidos e retornados pelos nossos motores de processamento.

Apesar dessa escolha, cada coluna já possui o tipo de dado específico que será armazenado, para otimizar a gestão dessas informações tanto dentro da plataforma do Firebase, quanto diretamente em nossa aplicação.

O banco de dados é estruturado da seguinte maneira, a fim de suportar a gestão das informações dentro da plataforma:

- **Tabela *users***

- **fullname:** Armazena o nome completo do usuário.
- **email:** Guarda o endereço de e-mail do usuário.

- **document:** Pode armazenar o documento de identificação do usuário, como CPF.
- **noticeid:** Chave estrangeira que faz referência ao edital (*notice*) associado ao usuário.

- Tabela *tasks*

- **title:** Título da tarefa.
- **description:** Descrição detalhada da tarefa.
- **contentid:** Chave estrangeira que referencia o conteúdo (*content*) associado à tarefa.
- **difficultyid:** Chave estrangeira que referencia o nível de dificuldade da tarefa.
- **hasfinished:** Indica se a tarefa foi concluída ou não.
- **userid:** Chave estrangeira que faz referência ao usuário que criou a tarefa.
- **dayofweek:** Dia da semana em que a tarefa deve ser realizada.
- **startat:** Horário de início da tarefa.
- **finishat:** Horário de término da tarefa.

- Tabela *difficulties*

- **name:** Nome do nível de dificuldade.
- **displayname:** Nome de exibição do nível de dificuldade.

- Tabela *subjects* (matérias)

- **name:** Nome da matéria.
- **noticeid:** Chave estrangeira que faz referência ao edital (*notice*) associado à matéria.

- Tabela *contents*

- **subjectid:** Chave estrangeira que referencia a matéria (*subject*) associada ao conteúdo.
- **text:** Texto do conteúdo, que pode conter informações relevantes para estudo.

- **Tabela notices (editais)**

- **name:** Nome do cargo ou edital.
- **filesrc:** Caminho para o arquivo do edital.
- **userid:** Usuário que fez o upload do edital.

Este modelo de banco de dados é projetado para permitir a associação de usuários a tarefas específicas, associadas a conteúdos de estudo e matérias específicas relacionadas aos editais. A inclusão de um nível de dificuldade (na tabela *difficulties*) proporciona uma maneira de classificar a complexidade das tarefas, enquanto a tabela notices permite o armazenamento e acesso aos editais relacionados aos estudos.

4.2.4 Integrações

Para a interpretação dos conteúdos pragmáticos dentro dos editais, faremos uso dos modelos de inteligência artificial fornecidos pela OpenAI. A OpenAI é uma organização de pesquisa em IA conhecida por seus avançados modelos de IA, especialmente na área de *Natural Language Processing* (NLP). Iremos utilizar esses modelos para desenvolver os seguintes motores:

- **Interpretação de Conteúdo de Edital e Geração de Matérias**

Este motor é encarregado de interpretar o conteúdo pragmático após a filtragem do edital fornecido pelo usuário. Ele irá identificar e extrair as matérias que serão cobradas no concurso, populando assim o banco de dados.

- **Interpretação de Matérias e Geração de Tarefas e Rotinas de Estudo**

Com o banco de dados já contendo as matérias identificadas, este motor entra em ação para gerar tarefas e rotinas de estudo personalizadas. Recebendo como entrada as matérias e conteúdo que o usuário ainda precisa estudar, ele irá gerar um cronograma de estudo detalhado, distribuindo as tarefas ao longo da semana de acordo com as necessidades e preferências do usuário.

Essas integrações permitem uma abordagem mais eficiente e personalizada no processo de estudo para concursos, aproveitando o poder dos modelos de linguagem avançados fornecidos pela OpenAI.

4.2.5 Versionamento de Código

O versionamento de código é uma prática fundamental no desenvolvimento de software, permitindo o controle e gerenciamento das alterações feitas ao longo do tempo

em um projeto. Para isso, utilizaremos o *GitHub* como plataforma de versionamento, que oferece uma série de recursos poderosos para colaboração e controle de versões. No nosso ambiente de desenvolvimento, teremos um repositório principal hospedado no *GitHub*:

Repositório do *Front-end* e *Functions* do Firebase:

Este repositório conterá o código-fonte do *Front-end* desenvolvido com *Next.js*, bem como as *Functions* do Firebase utilizadas no *Back-end*. Será organizado de forma a separar claramente os diretórios relacionados ao *Front-end* e às *Functions* do Firebase, mantendo uma estrutura de pastas intuitiva e coesa.

Estratégia de Versionamento: Gitflow

Para gerenciar as diferentes etapas de desenvolvimento e garantir uma colaboração eficiente entre os membros da equipe, adotaremos a estratégia de versionamento *Gitflow*. Essa abordagem define um modelo de fluxo de trabalho baseado em *branches*, que facilita a organização das funcionalidades em desenvolvimento, testes e produção. Principais *Branches*:

- **Main:** Esta *branch* representa a versão estável e de produção do código. Todo o código que está pronto para ser implantado em ambiente de produção é mesclado nesta *branch*.
- **Develop:** Esta *branch* é onde o desenvolvimento ativo ocorre. É a *branch* de integração para novas funcionalidades e correções de bugs. Todo o desenvolvimento é feito a partir desta *branch*.
- **Feature Branches:** Para cada nova funcionalidade ou tarefa, uma nova *branch* de *feature* é criada a partir da *branch develop*. Esta *branch* é utilizada para implementar a funcionalidade de forma isolada, antes de ser integrada de volta à *branch develop*.

Adotando essa estratégia de versionamento com o *Gitflow*, garantimos um desenvolvimento organizado, facilitando a colaboração entre os membros da equipe e mantendo um histórico claro e estruturado das alterações feitas no código-fonte ao longo do tempo.

4.2.6 Infraestrutura

Para hospedagem, optaremos por utilizar os serviços especializados de hospedagem da Vercel para o *Front-end* e do Firebase para o *Back-end* customizado.

Hospedagem do Front-end

A Vercel oferece um serviço de hospedagem altamente escalável e otimizado para aplicações *Front-end*, como o nosso desenvolvido com *Next.js*. Utilizando a plataforma da Vercel, podemos implantar e hospedar facilmente nosso *Front-end*, garantindo uma experiência de usuário rápida e confiável.

Hospedagem do *Back-end* no Firebase

O Firebase oferece por padrão a hospedagem de seus serviços diretamente em sua plataforma, eliminando a necessidade de recorrer a soluções terceirizadas para essa finalidade. Essa integração nativa proporciona uma infraestrutura completa e integrada, capaz de suportar todas as necessidades de nossa aplicação de forma eficiente e escalável.

4.2.7 Escalabilidade

Tanto a Vercel quanto o Firebase oferecem opções de escalabilidade conforme as necessidades do projeto. No caso da Vercel, podemos facilmente escalar nossa aplicação *Front-end* de acordo com o aumento da demanda de tráfego. Já o Firebase, além de oferecer hospedagem escalável, também permite dimensionar automaticamente o banco de dados e outros serviços conforme necessário.

4.2.8 Convenções e Padronização de Código

Convenções são acordos ou regras estabelecidas para padronizar a forma como realizamos determinadas atividades ou interações. No contexto do desenvolvimento de *software*, as convenções de codificação são diretrizes estabelecidas para padronizar a escrita e a organização do código-fonte de uma aplicação. Elas definem como devemos nomear variáveis, formatar o código, documentar funcionalidades e adotar certas práticas de desenvolvimento.

Para esse projeto, iremos seguir com as seguintes convenções e padrões de código:

- Nomenclatura de Variáveis e Funções:

Utilize nomes descritivos e significativos para variáveis e funções. Prefira *camelCase* para nomes de variáveis e funções em *JavaScript/TypeScript*.

- Comentários: Inclua comentários claros e concisos para explicar trechos de código complexos ou de difícil compreensão. Evite comentários óbvios que apenas repetem o que o código faz.

- Indentação e Formatação:

Utilize uma tabulação consistente para indentação, preferencialmente com 2 ou 4 espaços. Mantenha linhas de código com até 80-100 caracteres para facilitar a leitura em telas menores. Organize o código de forma clara e coesa, utilizando espaços em branco para separar blocos lógicos.

- Tratamento de Erros:

Sempre inclua tratamento de erros adequado em pontos críticos do código. Utilize *try-catch* para capturar e lidar com exceções de forma apropriada.

- Gerenciamento de Dependências:

Mantenha uma lista atualizada de todas as dependências e suas versões no arquivo de manifesto (como *package.json*). Utilize um gerenciador de dependências confiável, como [NPM](#) ou [YARN](#), e evite adicionar dependências desnecessárias.

- Revisões de Código:

Realize revisões de código regulares entre os membros da equipe para identificar e corrigir problemas de qualidade, estilo e desempenho. Mantenha um ambiente colaborativo e aberto para sugestões e melhorias no código.

Estabelecemos essas convenções com foco em garantir a consistência, legibilidade e manutenibilidade do projeto, pois elas não apenas facilitam o trabalho em equipe, permitindo que todos os desenvolvedores compreendam e contribuam com o código de maneira eficiente, mas também minimizam a ocorrência de erros e melhoram a qualidade geral do software.

4.3 Análise de Requisitos

Neste tópico é analisado e definido fatores de negócio do projeto, como requisitos funcionais, requisitos não funcionais, regras de negócios e histórias de usuários. Todos esses fatores foram levados em conta para o desenvolvimento do projeto e devem ser seguidos de forma objetiva e concreta.

4.3.1 Regras de Negócio

As regras de negócio são definições importantes para qualquer sistema e especificam o que é o escopo do sistema. No Quadro 4 estão especificadas as regras de negócio do Studyflow.

Quadro 4 – Regras de Negócio

ID	Descrição
RN1	Ao acessar o site, para ter acesso aos conteúdos e funcionalidades, usuário precisa estar logado
RN2	Apenas usuários autenticados podem executar as funcionalidades do sistema
RN3	Após o login bem-sucedido, o sistema deve redirecionar o usuário para a página principal da plataforma, onde ele terá acesso às funcionalidades disponíveis, tais quais, upload do edital
RN4	O sistema terá upload de arquivos para envio do edital ou um campo para digitar os conteúdos exigidos no edital
RN5	O sistema deve processar o arquivo enviado pelo usuário e extrair os dados relevantes, como disciplinas, conteúdos e datas de provas
RN6	Caso o formato do arquivo do edital enviado não seja suportado ou o conteúdo não seja identificado corretamente, o sistema deve fornecer feedback ao usuário sobre o problema
RN7	O sistema terá um calendário semanal onde os cards serão organizados
RN8	A rotina de estudos gerada deve ser baseada nas informações extraídas do edital do concurso e nas preferências de estudo do usuário, como disponibilidade de tempo e prioridades de aprendizado
RN9	Os cards semanais devem exibir informações detalhadas sobre os temas de estudo, incluindo descrição da matéria e tempo estimado de estudo
RN10	O usuário deve ser capaz de acessar a interface de edição dos cards semanais de tarefas a qualquer momento
RN11	As modificações feitas nos cards, como mover para outro dia da semana ou alterar a descrição da tarefa, devem ser refletidas instantaneamente na visualização da rotina de estudos
RN12	O sistema terá um plano pago onde as rotinas serão geradas sem limite e o número de editais será maior

Fonte: Os autores.

As regras de negócios definidas abrangem todo o escopo inicial para a plataforma

Studyflow. As principais regras de negócio que devem ser atendidas e com prioridade de desenvolvimento ou manutenção são as regras 5, 7 & 8. Ao elaborarmos o escopo do projeto, entendemos que essas são as principais funcionalidades, que se houver falha ou mudança na regra, terão grandes impactos na jornada e experiência de uso dos usuários.

4.3.2 Requisitos Funcionais

Em Engenharia de Software, os requisitos funcionais são a espinha dorsal de qualquer projeto, definindo as funcionalidades essenciais que o sistema deve oferecer para atender às necessidades dos usuários e alcançar os objetivos do projeto. Segundo (SOMMERVILLE, 2011), "requisitos funcionais especificam o que o sistema deve fazer". Eles descrevem as tarefas e atividades que o software deve realizar, traduzindo as expectativas dos stakeholders em funcionalidades concretas.

Para o Studyflow, os seguintes requisitos funcionais estão descritos do Quadro 5.

Quadro 5 – Requisitos Funcionais

ID	Descrição
RF1	Sistema de cadastro e login
RF2	O sistema deve permitir que o usuário faça upload de um arquivo contendo o edital do concurso desejado
RF3	O sistema deve permitir que o usuário digite o conteúdo a ser estudado exigido pelo edital
RF4	O sistema deve processar o arquivo enviado e extrair automaticamente as informações relevantes
RF5	O sistema deve fornecer uma interface para que o usuário visualize sua rotina de estudos semanalmente
RF6	A visualização da rotina de estudos deve incluir cards organizados por dia da semana, exibindo os temas de estudo, descrições das tarefas e tempos estimados de estudo
RF7	O sistema deve permitir que o usuário edite os cards de tarefas em sua rotina de estudos
RF8	A movimentação dos cards deve ser intuitiva e feita através de arrastar e soltar na interface da plataforma
RF9	O sistema deve solicitar feedback semanal ao usuário sobre os conteúdos estudados
RF10	O sistema deve ser capaz de gerar automaticamente uma rotina de estudos com base nas informações extraídas do edital do concurso e nas preferências do usuário
RF11	O sistema deve permitir que o usuário personalize suas preferências de estudo, como horários disponíveis e prioridades de aprendizado
RF12	As preferências de estudo personalizadas devem ser levadas em consideração na geração da rotina de estudos

Os requisitos funcionais definidos no quadro 5 são as bases para toda a construção do software, assim como as regras de negócio definidas anteriormente.

Neste primeiro momento de planejamento para o desenvolvimento, os requisitos funcionais 4, 6 & 10 são prioridades para a implementação.

Os requisitos citados anteriormente foram definidos como prioridade pela sua difícil implementação, já que demandam uma lógica de implementação mais complexa que os outros requisitos, além de envolver a manipulação de [IA](#).

4.3.3 Requisitos Não Funcionais

Garantindo a Qualidade do Software, os requisitos não funcionais complementam os requisitos funcionais, definindo as características e qualidades que o sistema deve apresentar para garantir sua usabilidade, confiabilidade, segurança, desempenho e outras características essenciais. Para ([PRESSMAN; MAXIM, 2014](#)), "requisitos não funcionais definem como o sistema deve se comportar". Eles detalham as restrições e os critérios que o software deve atender para oferecer uma experiência satisfatória aos usuários.

Quadro 6 – Requisitos Não Funcionais

ID	Descrição
RNF1	O sistema deve garantir a segurança dos dados do usuário, implementando medidas de criptografia para proteger informações sensíveis, como senhas e dados pessoais
RNF2	O acesso aos dados do usuário deve ser restrito apenas a usuários autorizados e ser protegido contra acessos não autorizados ou ataques maliciosos
RNF3	O sistema deve ser capaz de lidar com um grande volume de usuários simultâneos sem degradar significativamente o desempenho
RNF4	O tempo de resposta do sistema para as solicitações do usuário deve ser no máximo 3 segundos, garantindo uma experiência de uso fluida e sem atrasos perceptíveis
RNF5	A plataforma deve ser compatível com os principais navegadores web, como Google Chrome, Mozilla Firefox, Safari e Microsoft Edge, garantindo uma experiência consistente para todos os usuários
RNF6	A interface do usuário da plataforma deve ser responsiva e adaptável a diferentes dispositivos e tamanhos de tela, incluindo desktops, laptops, tablets e smartphones
RNF7	O sistema deve oferecer suporte para inglês e português

Assim como os requisitos funcionais e regras de negócio, os requisitos não funcionais também possuem prioridades a serem priorizadas no planejamento.

Pensando na experiência do usuário com a plataforma, o requisito não funcional 4 é uma das prioridades, por impactar a fluidez da jornada do usuário utilizando o Studyflow.

4.4 Histórias de Usuário

Pensando na jornada do usuário dentro do sistema, foram elaboradas 6 histórias de usuário, que inicialmente cobrem o fluxo de caminhos que os usuários podem fazer dentro da plataforma.

A escolha de escrever as histórias de usuário, foi feita de maneira conjunta, entendendo que é a forma mais fácil de representar as jornadas e fluxos para qualquer pessoa que queira entender como o Studyflow funcione.

1. Realizar Cadastro

Caso: Como usuário interessado em utilizar a plataforma de estudos, desejo me cadastrar no sistema para ter acesso aos recursos disponíveis.

Critérios de Aceitação:

- O usuário deve conseguir preencher o formulário de cadastro.
- O e-mail fornecido pelo usuário deve ser único no sistema.
- A senha deve atender aos padrões de segurança estabelecidos pelo site.

2. Realizar Login

Caso: Como usuário registrado, desejo fazer login na plataforma para acessar os recursos disponíveis.

Critérios de Aceitação:

- O usuário deve conseguir fazer login com sucesso utilizando e-mail e senha corretos.
- O acesso aos recursos da plataforma só deve ser permitido para usuários logados.

3. Upload de Edital para Concurso

Caso: Como usuário da plataforma de estudos, desejo poder fazer o upload de um edital de concurso para que a plataforma possa gerar uma rotina de estudos personalizada.

Critérios de Aceitação:

- O usuário deve conseguir fazer o upload de um arquivo contendo o edital do concurso desejado.
- A plataforma deve ser capaz de processar o edital e extrair os conteúdos relevantes para a preparação.
- Os conteúdos extraídos devem ser organizados de forma a criar uma rotina de estudos semanal.

4. Gerar Rotina Semanal de Estudos

Caso: Como usuário da plataforma, desejo que a plataforma gere uma rotina semanal de estudos com base no conteúdo do edital do concurso que fiz o upload.

Critérios de Aceitação:

- A plataforma deve analisar o edital e propor uma rotina de estudos que cubra todos os temas e disciplinas exigidos.
- A rotina deve ser organizada em cards em um calendário semanal, indicando os temas a serem estudados a cada dia.

- Os cards devem ser movíveis para permitir ao usuário ajustar a rotina de acordo com sua disponibilidade e preferências.

5. Coleta de Feedback Semanal

Caso: Como usuário da plataforma de estudos, desejo fornecer *feedback* semanal sobre os conteúdos estudados para que a plataforma possa ajustar a rotina de estudos.

Critérios de Aceitação:

- Ao final de cada semana, o usuário deve ser solicitado a fornecer *feedback* sobre os conteúdos estudados.
- O *feedback* deve incluir informações sobre as matérias estudadas e a dificuldade encontrada em cada uma.
- A plataforma deve usar o *feedback* fornecido pelo usuário para ajustar a rotina de estudos para a próxima semana, priorizando as matérias com maior dificuldade.

4.5 Fases de Entrega

As fases de entrega de um projeto são etapas fundamentais que conduzem o desenvolvimento de uma solução desde sua concepção até sua implementação e lançamento. Cada fase representa um marco importante no ciclo de vida do projeto, onde objetivos específicos são alcançados e progresso significativo é realizado.

Nossas fases de entrega planejadas são:

- Planejamento e Análise: Nesta fase, são identificados os requisitos do projeto, definidos os objetivos e escopo, e elaborado o plano de projeto detalhado. Também é realizada uma análise de viabilidade técnica e financeira.
- Prova de Conceito (PoC):
A *Proof of Concept* (PoC) é uma fase inicial do projeto na qual são desenvolvidos protótipos ou demonstrações que validam a viabilidade técnica das principais funcionalidades da plataforma. Nesta fase, focamos em implementar um conjunto mínimo de recursos para validar a solução proposta e demonstrar sua viabilidade.
- Desenvolvimento Iterativo: Após o sucesso da PoC, o desenvolvimento da plataforma começa em etapas iterativas. Funcionalidades adicionais são implementadas em ciclos de desenvolvimento curtos, permitindo feedback contínuo e ajustes conforme necessário.

- **Testes e Qualidade:** Durante todo o processo de desenvolvimento, são realizados testes rigorosos para garantir que a plataforma atenda aos requisitos de qualidade e segurança. Isso inclui testes de unidade, integração, aceitação do usuário e segurança.
- **Implantação e Lançamento:** Após a conclusão do desenvolvimento e dos testes, a plataforma é implantada em um ambiente de produção e está pronta para ser lançada. Isso pode incluir a configuração de servidores, migração de dados e treinamento de usuários.
- **Monitoramento e Manutenção:** Após o lançamento, a plataforma é continuamente monitorada para garantir que esteja funcionando conforme o esperado. Também são feitas atualizações regulares e manutenção para corrigir *bugs*, adicionar novos recursos e melhorar a experiência do usuário.

Prova de Conceito (PoC) Inicial: Na PoC inicial, nosso foco seria desenvolver um protótipo funcional da plataforma que demonstre as principais funcionalidades propostas. Isso pode incluir a autenticação de usuários, a criação de tarefas e a integração com serviços de terceiros, como o Firebase para o *back-end* e a Vercel para o *front-end*. A PoC permite validar a viabilidade técnica da solução proposta e identificar possíveis desafios ou obstáculos que precisam ser superados antes da implementação completa.

Entrega Final da Plataforma:

Na entrega final da plataforma, todos os recursos e funcionalidades planejados são implementados e testados completamente. Isso inclui a implementação completa e treinamento dos modelos de IA, adequação a políticas de segurança, a integração com serviços de terceiros, como autenticação, armazenamento de dados e hospedagem, e a garantia de que a plataforma atenda aos requisitos de desempenho, escalabilidade e usabilidade. A entrega final marca o lançamento oficial da plataforma e seu uso pelos usuários finais.

4.6 Segurança

A segurança da aplicação é uma preocupação em qualquer sistema de software, especialmente quando se trata de lidar com dados sensíveis dos usuários. A proteção dessas informações é fundamental não apenas para garantir a confiança dos usuários, mas também para cumprir regulamentações de privacidade e evitar violações de dados.

O projeto contara com usuários reais, o que implica na captação e armazenamento de dados. No Brasil, qualquer empresa está sujeita a regulações de proteção de dados dos usuários e como a [Lei Geral de Proteção de Dados \(LGPD\)](#), que estabelece diretrizes

claras sobre como as organizações devem lidar com os dados pessoais dos usuários, impondo penalidades rigorosas para aqueles que não cumprem os requisitos de segurança e privacidade. O cenário apontado por (JATOBA, 2021), 84% das empresas brasileiras não estão preparadas para atender as especificidades da LGPD.

A maior preocupação das empresas, o que inclui o Styudflow, são as sanções financeiras que a LGPD prevê. As multas são de 2% do faturamento anual e limitada a R\$ 50 milhões.

Com a finalidade de cumprir a regulamentação e a proteção do usuário e do Studyflow, serão propostos termos de uso para que haja segurança em ambas as partes.

4.6.1 Autenticação e Autorização

A autenticação e autorização são pilares fundamentais da segurança da aplicação, desempenhando papéis essenciais na proteção dos dados e na prevenção de acessos não autorizados. Vamos explorar como esses conceitos são implementados para garantir a segurança da nossa plataforma.

Autenticação: A autenticação é o processo pelo qual a identidade de um usuário é verificada, garantindo que apenas usuários legítimos tenham acesso aos recursos da aplicação. Para isso, adotaremos um sistema robusto de autenticação baseado em *JSON WEB Tokens* (JWT). Quando um usuário faz login na plataforma, suas credenciais são verificadas e, se válidas, um token JWT é gerado e enviado de volta ao cliente. Esse token contém informações sobre a identidade do usuário e é assinado digitalmente para garantir sua autenticidade.

O Firebase desempenha um papel crucial na interação com os tokens JWT, facilitando a autenticação segura dos usuários em nossa aplicação. Aqui está um resumo de como o Firebase interage com os tokens:

- **Geração de Tokens JWT:** Quando um usuário realiza o processo de autenticação no Firebase, seja por e-mail e senha, login social (como *Google* ou *Facebook*) ou outros métodos suportados, o Firebase gera um token JWT válido que contém informações sobre a identidade do usuário, como seu ID exclusivo, endereço de e-mail e quaisquer metadados adicionais necessários para autorização.
- **Validação dos Tokens:** O Firebase verifica a validade dos tokens JWT antes de conceder acesso aos recursos protegidos. Isso inclui a verificação da assinatura digital do token para garantir sua autenticidade e a verificação de outros atributos, como a data de expiração, para garantir que o token ainda esteja válido.

- **Decodificação dos Tokens:** O Firebase decodifica os tokens [JWT](#) para extrair as informações relevantes sobre o usuário. Isso pode incluir o [ID](#) do usuário, endereço de e-mail e quaisquer outros dados personalizados incluídos no token durante a geração.
- **Gerenciamento de Sessões:** O Firebase gerencia automaticamente as sessões de autenticação dos usuários, mantendo os tokens [JWT](#) válidos e atualizados durante o período de uso da aplicação. Isso permite que os usuários permaneçam autenticados em diferentes dispositivos e sessões sem a necessidade de reautenticação frequente.
- **Controle de Acesso:** Com base nas informações contidas nos *tokens* [JWT](#), o Firebase aplica políticas de autorização para determinar quais recursos e operações o usuário autenticado tem permissão para acessar. Isso inclui a aplicação de regras de segurança personalizadas em serviços como o *Firebase Realtime Database* e o *Cloud Firestore*.

Em resumo, o Firebase oferece uma integração completa e segura com *tokens* [JWT](#), facilitando a autenticação e autorização dos usuários em nossa aplicação. Essa abordagem robusta garante a proteção dos dados dos usuários e mantém a segurança da aplicação em todos os níveis.

Autorização: A autorização, por sua vez, determina quais recursos e operações um usuário autenticado tem permissão para acessar e executar. Para isso, implementaremos controles de acesso granulares em toda a plataforma. Cada solicitação feita por um usuário será verificada em relação às políticas de autorização configuradas. Isso inclui restrições de acesso a determinadas funcionalidades com base no papel do usuário, como administrador, moderador ou usuário padrão.

Ao implementar um sistema de autenticação e autorização sólido, estamos fortalecendo as defesas da nossa aplicação contra ameaças de segurança. O uso de *tokens* [JWT](#) oferece uma camada adicional de segurança, pois eles são assinados digitalmente e podem incluir informações sobre o usuário, como permissões e papéis, que são verificados em cada solicitação.

Além disso, ao adotar práticas de segurança recomendadas, como criptografia de dados, gerenciamento seguro de credenciais e monitoramento de atividades suspeitas, estamos reduzindo significativamente o risco de violações de segurança e protegendo os dados sensíveis dos nossos usuários.

4.6.2 Políticas de Segurança

As políticas de segurança desempenham um papel vital na proteção dos ativos e dados de uma plataforma contra ameaças cibernéticas. Elas estabelecem diretrizes e

regras para garantir a confidencialidade, integridade e disponibilidade dos dados, bem como para prevenir acessos não autorizados e mitigar riscos de segurança.

No contexto da nossa plataforma, as políticas de segurança são aplicadas em várias áreas. Implementamos políticas de controle de acesso e identidade para garantir que apenas usuários autorizados tenham acesso aos recursos e funcionalidades da plataforma. Isso inclui a utilização de autenticação forte, controle de acesso baseado em função e monitoramento de atividades suspeitas.

Além disso, estabelecemos políticas de criptografia para proteger os dados sensíveis dos usuários durante o armazenamento e a transmissão. Os dados são criptografados de forma a garantir que somente as partes autorizadas tenham acesso às informações, mesmo que ocorra uma violação de segurança.

Outro aspecto importante é a implementação de políticas de monitoramento e resposta a incidentes. Realizamos monitoramento contínuo dos sistemas e registros de atividades para detectar e responder rapidamente a possíveis ameaças. Isso inclui a realização de auditorias de segurança regulares e a elaboração de planos de resposta a incidentes para lidar com situações de emergência.

Em resumo, as políticas de segurança são essenciais para proteger os dados dos usuários e garantir a confiabilidade e integridade da nossa plataforma. Ao implementar políticas de segurança robustas e alinhadas às melhores práticas do setor, estamos comprometidos em oferecer um ambiente seguro e confiável para nossos usuários.

4.7 Viabilidade Financeira

Inicialmente, para a fase de desenvolvimento e testes, vamos utilizar os planos gratuitos oferecidos pela Vercel para hospedar nosso *front-end*. Essa escolha nos permitirá iniciar o projeto de maneira ágil e econômica, facilitando a implantação e os testes iniciais da aplicação.

No entanto, para a fase de produção e lançamento oficial da plataforma, planejamos migrar para soluções pagas. Na Vercel, iremos adotar, inicialmente, o plano Pro, que oferece recursos adicionais, como escalabilidade aprimorada, limites maiores de *bandwidth* e de cachê de dados, além de suporte especializado. Isso garantirá um desempenho consistente e confiável da nossa aplicação em ambiente de produção, além de proporcionar um nível mais alto de serviço e suporte.

Já no Firebase, desde a fase de desenvolvimento iremos optar pelo plano pago Blaze. Isso se dá pois esse plano nos dá acesso a serviços específicos da plataforma, como as Functions, funcionalidade que será essencial para o funcionamento da plataforma. Apesar de ser um plano pago, nossos custos iniciais serão baixos, pois o plano segue o formato

"*pay-as-you-go*", ou seja, o faturamento da conta será conforme o uso da plataforma. A flexibilidade desse modelo de pagamento nos permitirá começar com custos mínimos e aumentar conforme o crescimento e a demanda da aplicação.

Essas abordagens nos possibilitam começar com investimentos mínimos durante a fase de desenvolvimento, ajustando nossos gastos de acordo com o crescimento e a maturidade do projeto. Dessa forma, garantimos uma transição suave para o ambiente de produção, maximizando o valor entregue aos usuários finais e garantindo o sucesso contínuo da nossa aplicação.

4.7.1 Custos

Até a criação da aplicação não haverá custos com as tecnologias já citadas. Para a próxima fase da aplicação, o teste de mercado, os custos de se manter a aplicação de pé e até 10 mil requisições foram levantados nas tabelas 2 e 3, que tratam dos custos fixos e variáveis respectivamente.

Tabela 2 – Custos Fixos

Tecnologias	Custo Mensal
Firestore	15.00
Cloud Storage	10.00
Cloud Functions	30.00
Vercel	100.00
Serviço de e-mail	40.00
Domínio	45.00

Fonte: Os autores.

Os custos variáveis são todos referentes a [API](#) da OPenAI. Para cada modelo utilizado o custo de requisição se difere.

Tabela 3 – Custos Variáveis

Tasks IA	Custo por requisição	Requisições mensais	Total Mês
Leitura de Edital	0.05	10000	500.00
Geração de Tarefas	0.02	10000	200.00

Ao total, com os custos fixos e variáveis, para a plataforma do Studyflow se manter no ar e funcionando, com a capacidade de até 10 mil requisições mensais, é necessário a quantia de R\$ 940,00 mensais.

4.7.2 Planos de Assinatura e Expectativa Financeira

Após a análise dos custos envolvidos na operação da plataforma, reconhecemos a necessidade de implementar um modelo de assinatura para garantir o acesso completo às

ferramentas oferecidas pela nossa aplicação. Este modelo de assinatura será oferecido em formatos mensal e anual, proporcionando flexibilidade aos usuários de acordo com suas preferências e necessidades. Abaixo, detalhamos os planos disponíveis:

Plano Gratuito

Com o plano gratuito, o usuário terá acesso a todas as funcionalidades da plataforma. Porém, ele está limitado a fazer o upload de apenas um edital, e de utilizar a geração de tarefas por até 3 semanas. Dessa forma, o usuário pode ter uma experiência dentro da plataforma, e decidir se faz sentido ou não começar a pagar pelos serviços completos.

Plano Pago

Nosso Plano Pago oferece aos usuários acesso ilimitado a todas as funcionalidades e recursos avançados da plataforma. Com este plano, os usuários podem fazer o upload de múltiplos editais e aproveitar a geração ilimitada de tarefas.

Para cobrir os custos operacionais e garantir a sustentabilidade da plataforma, estimamos o preço mensal da assinatura em R\$24,99 e o preço anual em R\$249,90. Essa estrutura de preços proporciona aos usuários flexibilidade e economia, incentivando-os a aderir ao plano anual para obter um desconto significativo.

4.8 Prova de Conceito (POC)

Na [PoC](#), o objetivo é o teste da arquitetura e tecnologias propostas anteriormente, com a intenção de verificar se os componentes se integram e o sistema pensado é possível de ser desenvolvido.

4.8.1 Mudanças

Nosso objetivo ao desenvolver a [PoC](#) era testar e validar o funcionamento das tecnologias e da infraestrutura planejadas inicialmente. Para isso, implementamos as integrações essenciais utilizando o Firebase, como login e registro, além de uma integração básica com o serviço de pagamento Stripe. Também desenvolvemos o motor de processamento e identificação dos conteúdos e matérias de um edital dentro da plataforma, junto das funcionalidades no frontend para suportar esse recurso.

4.9 Mínimo Produto Vável (MVP)

Esta seção tem por objetivo informar as alterações, descartes e escolhas feitas no projeto para o MVP, em relação ao sistema da [PoC](#). Essas mudanças refletem os problemas

e inconsistências enfrentadas durante o desenvolvimento da PoC, e foram motivadas pela necessidade de otimizar a eficiência e o desempenho do sistema, de forma a atender os requisitos do MVP da melhor maneira possível.

4.9.1 Mudanças na Infraestrutura

A principal atualização feita no projeto em relação à PoC foi o descarte do Google Gemini, substituído pelo serviço da OpenAI. Inicialmente, o Gemini foi escolhido devido ao seu baixo custo e à integração facilitada com o Firebase, visando manter todos os serviços em um único ecossistema. Além disso, o Gemini oferecia um serviço de treinamento de modelos customizados, baseado em outros modelos de processamento de texto pré-treinados da Google.

Um dos primeiros problemas enfrentados com o serviço do Gemini foi justamente a customização dos modelos. Embora tivéssemos sucesso em treinar e utilizar um modelo com textos de editais e retorno de JSON, não conseguimos utilizar o mesmo fora da plataforma da Google, ou seja, não conseguimos integrar o serviço ao nosso código. Além disso, por ser um serviço relativamente recente, a documentação em certas funcionalidades ainda é escassa, assim como o suporte da comunidade.

Os problemas que mais nos afetaram foram a inconsistência e os bloqueios ao utilizar um modelo de processamento de texto já treinado. Os textos JSON gerados pela IA eram frequentemente quebrados ou não formatados corretamente, impossibilitando seu consumo dentro da plataforma. Além disso, ao enviar o mesmo texto do edital diversas vezes para processamento, o sistema retornava erros de "recitação" ou de conteúdo não autorizado. Esses erros também possuíam uma documentação extremamente limitada, dificultando ainda mais a solução dos problemas.

Essas dificuldades nos levaram a substituir o Google Gemini pelo serviço da OpenAI, que oferecia maior estabilidade e consistência na geração dos conteúdos, melhor documentação e suporte mais robusto. A substituição dos serviços foi um processo relativamente simples, e desde o início os modelos da OpenAI se mostraram muito eficientes. Porém, tivemos a mudança nos custos também, anteriormente, pois estávamos trabalhando com os serviços gratuitos do Gemini, e agora temos outros valores e custos consideravelmente maiores para levar em consideração para o projeto no futuro.

A mudança do serviço de IA utilizado foi a única alteração relevante nesse período. O restante do sistema e outras integrações se mostraram satisfatórios, atendendo às expectativas e requisitos estabelecidos durante a fase de planejamento. A plataforma do Studyflow pode ser acessada através do link <<https://studyflow-three.vercel.app>> ou através do QR Code da figura 5.

No processo de desenvolver o MVP, prezamos pela otimização de custos, como

Figura 5 – QRCode do Deploy



Fonte: Os autores.

citado anteriormente, as requisições para a [API](#) do GPT tem um custo, por isso prezamos por testar e fazer apenas requisições necessárias. Como melhorias futuras, o estudo da equipe é como fazer o mínimo de requisições possíveis ao sistema externo, reaproveitando dados das requisições que foram feitas em algum momento.

5 Considerações Finais

Com as pesquisas e experimentações sistêmicas feitas ao longo do semestre, enxergamos potencial na aplicação Studyflow. Dentre as implementações que foram feitas, entendemos que a utilização da Inteligência Artificial foi um grande diferencial no desenvolvimento, sendo também um dos pontos que mais requiriu atenção da equipe.

Para ([ESTANISLAU, 2019](#)), a utilização de recursos tecnológicos ajuda os estudantes a obterem melhores resultados em sua formação. A proposta da plataforma vai de encontro com os estudantes e referências utilizadas ao longo do texto, de que a união de métodos e tecnologia potencializa o resultado dos estudantes e no nosso caso, potencializa a chance do usuário a ser aprovado em um concurso.

Referências

- BASTOS, A. A busca por concursos públicos cresce 402019. Disponível em: <<https://www.jusbrasil.com.br/artigos/a-busca-por-concursos-publicos-cresce-40-ao-ano-de-acordo-com-dados-da-anpac/704008054>>. Citado na página 11.
- ESTANISLAU, L. J. M.; MORAIS, A. P. P.; CABRAL, R. O.; SOBRINHO, J. E. d. L.; LORENA, S. B. d. et al. Uso de aplicativos de tecnologia móvel na rotina de estudantes concluintes de medicina. 2019. Citado na página 48.
- IAHN, L. F.; MAGALHÃES, L. E. R.; BENTES, R. d. F.; GERAL, E. C. em. Educação a distância x educação presencial: estudo comparativo entre dois cursos preparatórios para concurso. In: *CONGRESSO ABED*. [S.l.: s.n.], 2008. p. 1–13. Citado na página 11.
- JATOBA, M. *LCPD, empresas nao estao preparadas*. 2021. Disponível em: <<https://www.folhape.com.br/economia/lcpd-84-das-empresas-nao-estao-preparadas/184486>>. Citado na página 41.
- LEITE, U. do R.; TAMAYO, Á.; GÜNTHER, H. Organização do uso do tempo e valores de universitários. *Avaliação Psicológica: Interamerican Journal of Psychological Assessment*, Instituto Brasileiro de Avaliação Psicológica (IBAP), v. 2, n. 1, p. 57–66, 2003. Citado na página 16.
- NETTO, H. G. da C.; SOUZA, R. N. P. M. de. A importância da educação a distância na formação do conhecimento para os concursos públicos. *Brazilian Journal of Development*, v. 5, n. 6, p. 4590–4598, 2019. Citado na página 11.
- PRESSMAN, R.; MAXIM, D. B. R. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education, 2014. ISBN 9780078022128. Disponível em: <<https://books.google.com.br/books?id=i8NmnAEACAAJ>>. Citado na página 36.
- SALGADO, L. *Tem dificuldades na hora de estudar para concursos? Saiba como melhorar*. 2016. Disponível em: <<https://g1.globo.com/economia/concursos-e-emprego/blog/tira-duvidas-de-concursos/post/tem-dificuldades-na-hora-de-estudar-para-concursos-saiba-como-melhorar.html>>. Citado 2 vezes nas páginas 11 e 16.
- SANT'ANA, J. *Governo federal anuncia quase 6 mil vagas para concursos públicos neste ano; confira os cargos*. 2023. Disponível em: <<https://valorinveste.globo.com/mercado-s/brasil-e-politica/noticia/2023/06/16/governo-federal-anuncia-mais-de-44-mil-vagas-para-concursos-publicos-neste-ano-confira-os-cargos.ghml>>. Citado na página 11.
- SANTOS, I. A. dos. Métodos e técnicas de estudo que potencializam o ensino-aprendizagem. *Universidade de Brasília*, 2015. Citado na página 16.
- SHEMSHACK, A.; SPECTOR, J. M. A systematic literature review of personalized learning terms. *Smart Learning Environments*, Springer, v. 7, n. 1, p. 33, 2020. Citado na página 17.

SILVA, A. C. B. et al. Inteligência artificial no processo de aprendizagem: softwares usados para complementar o conhecimento e auxiliar a rotina de estudos. Insitituto Federal de Educação, Ciência e Tecnologia de Goiás, 2023. Citado na página 17.

SOMMERVILLE, I. Software engineering (ed.). *America: Pearson Education Inc*, 2011. Citado na página 35.