



WELLYNGTON DE JESUS ARRUDA JUNIOR
WESLEY RODRIGUES GUIMARÃES

ALFAEDU: DESENVOLVIMENTO DE SOFTWARE PARA AUXÍLIO AOS
PROFESSORES ALFABETIZADORES

Ji-Paraná

2021

WELLYNGTON DE JESUS ARRUDA JUNIOR
WESLEY RODRIGUES GUIMARÃES

ALFAEDU: DESENVOLVIMENTO DE SOFTWARE PARA AUXÍLIO AOS
PROFESSORES ALFABETIZADORES

Monografia apresentada à Banca Examinadora do Curso Sistemas de informação do Centro Universitário São Lucas, como requisito de aprovação para obtenção do Título de Bacharel em Sistemas de Informação.

Orientadora Profa. Me.: Ana Flávia Moreira Camargo.

Ji-Paraná
2021

Dados Internacionais de Catalogação na Publicação - CIP

A779a Arruda Junior, Wellyngton de Jesus.

AlfaEdu: desenvolvimento de software para auxílio aos professores alfabetizadores. / Wellyngton de Jesus Arruda Junior ; Wesley Rodrigues Guimarães. – Ji-Paraná, 2021.
65 p., il.

Monografia (Curso de Sistemas de Informação) – Centro Universitário São Lucas Ji-Paraná, 2021.

Orientadora: Me. Ana Flávia Moreira Camargo

1. Desenvolvimento de software. 2. Educação. 3. Alfabetização - tecnologia. 4. Software. I. Guimarães, Wesley Rodrigues II. Camargo, Ana Flávia Moreira. III. Título.

CDU 004.4:37

WELLYNGTON DE JESUS ARRUDA JUNIOR

WESLEY RODRIGUES GUIMARÃES

**ALFAEDU: DESENVOLVIMENTO DE SOFTWARE PARA AUXÍLIO AOS
PROFESSORES ALFABETIZADORES**

Monografia apresentada à Banca Examinadora do Curso Sistemas de informação do Centro Universitário São Lucas, como requisito de aprovação para obtenção do Título de Bacharel em Sistemas de Informação.

Orientadora Profa. Me.: Ana Flávia Moreira Camargo.

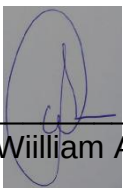
Ji-Paraná, 23 de junho de 2021.

Avaliação/Nota: Trabalho de Conclusão de Curso (X) aprovado ou ()reprovado com nota total de 8,6 (oito virgula seis) pontos, sendo atribuídos o valor 8,6 (oito virgula seis) ao trabalho escrito e 8,7(oito virgula sete) à apresentação oral.

Wellyngton de Jesus Arruda Junior Wesley Rodrigues Guimarães

WELLYNGTON DE JESUS ARRUDA JUNIOR E WESLEY RODRIGUES
GUIMARÃES

BANCA EXAMINADORA


Prof. Esp. William Alves Fachetti

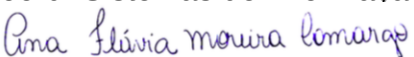
Centro Universitário São Lucas Ji-Paraná


Prof. Esp. José Rodolfo Milazzotto Olivas

Centro Universitário São Lucas Ji-Paraná


Prof. Me. Thyago Bohrer Borges
Coord. Sistemas de Informação

Centro Universitário São Lucas Ji-Paraná


Prof. Me. Ana Flavia Moreira Camargo

Centro Universitário São Lucas Ji-Paraná

Orientadora

RESUMO

Vivemos em um país em que a educação está deficiente e cada dia mais necessita de novas formas de ensino, sabendo que a educação infantil principalmente a alfabetização é a parte mais importante na educação, pois é quando o aluno recebe toda a base da educação e tem acesso a novos conhecimentos, muitas escolas no país utilizam alguns *softwares* e jogos para o auxílio na alfabetização, com base nisso, junto a experiência de um dos autores em uma escola que utilizava-se desses programas, analisamos esses *softwares* e percebemos a falta de um *feedback* ao professor, pois os *softwares* existentes disponibilizavam apenas as atividades, diante desse cenário e utilizando-se de pesquisas sobre o funcionamento desses *softwares* e jogos, foi decidido que este trabalho tem como objetivo principal o desenvolvimento de um programa que auxilia o professor na alfabetização, onde ocorra um *feedback* imediato sobre as questões resolvida pelos alunos e crianças, esse *feedback* será disponibilizado tanto após o termino da atividade e se o professor optar ele poderá ser enviado via e-mail.

Palavras-chave: Educação. Alfabetização. *Software*.

ABSTRACT

We live in a country which the education is faulting and each day needs new ways of teaching, knowing that the children education mainly the literacy is the most important part in education, because it is when the student receive all the base of the education and have more access to new knowledges, many schools in the country use some softwares and games to help in the literacy, based on it together to the experience of one the author in a school that utilized one of theses programs, we have analized these softwares and noticed a lack of feedback to the teacher, because the softwares existing only provided the activities, as a consequence of this scenario and utilizing theses researchs about how theses softwares and games works, it was decided that this work has as mainly objective the developing of a program that help the teacher in the literacy, where occurs a imediatly feedback about these solved questions by the students and children, this feedback will be released after the end of an activity and if the teacher decides it can be sent by email.

Keywords: Education. Literacy. Softwares.

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de casos de uso	25
Figura 2 - Diagrama de classe	28
Figura 3 - Diagrama de sequência Cadastrar aluno	29
Figura 4 - Diagrama de sequência Escolher atividades	29
Figura 5 - Diagrama de sequência Fazer atividades	30
Figura 6 - Diagrama de atividades	31
Figura 7 - Diagrama de componentes	32
Figura 8 - Diagrama de entidade e relacionamento	32
Figura 9 - Código da primeira tela criada	48
Figura 10 - Código da imagem de feedback.....	49
Figura 11 - Código de envio do e-mail	50
Figura 12 - Tela de cadastro	51
Figura 13: Atividade clique na figura	52
Figura 14 - Atividade digite o nome da figura.....	53
Figura 15 - Atividade clique na letra que falta	53
Figura 16 - Tela de feedback.....	54
Figura 17 - Tela de login	55
Figura 18 - Tela de cadastro	55
Figura 19 - Tela de editar dados do aluno.....	56
Figura 20 - GitKrakem	57
Figura 21 - Discord.....	58
Figura 22 - Confirmação de senha	59
Figura 23 - Confirmação de e-mail	59
Figura 24 - Teste inserção do aluno no banco de dados	60
Figura 25 - Botão de editar aluno	60
Figura 26 - Feedback no email.....	61

LISTA DE TABELAS

Tabela 1 - Documento de requisitos Escolher atividade	22
Tabela 2 - Documento de requisitos Realizar atividade	23
Tabela 3 - Documento de requisitos Avaliação do educador	23
Tabela 4 - Documento de requisitos Efetuar Login	23
Tabela 5 - Documento de requisitos Realizar cadastro	24
Tabela 6 - Custo Total do Projeto.....	33
Tabela 7 - Equipe de desenvolvimento	33
Tabela 8 - Product backlog.....	34
Tabela 9 - Sprints backlog.....	35
Tabela 10 - Informações sobre as sprints	43

LISTA DE ABREVIATURAS E SIGLAS

Art	Artigo.
BNCC	Base Nacional Comum Curricular.
CLATES	Centro Latino-Americano de Tecnologia Educacional.
FBS	Fman Build System.
IDE	Integrated Development Environment.
IHC	Interface Homem Computador.
IOS	Iphone Operating System.
ISO	International Organization for Standardization
JPG	Joint Photographics Experts Group.
MEC	Ministério da Educação.
NCE	Núcleo de Cálculos Especiais
NUTES	Núcleo de Tecnologia Educacional para a Saúde.
OpenGL	Open Graphics Library.
PNA	Política Nacional de alfabetização.
PNG	Portable Network Graphics.
RF	Requisitos Funcionais.
RNF	Requisitos não funcionais.
SBP	Sociedade Brasileira de pediatria.
Sealf	Secretaria de Alfabetização.
SQL	Standard Query Language.
SVG	Scalable Vectorial Graphics.
TDIC	Tecnologias Digitais de Informação e Comunicação .
USP	Universidade de São Paulo.
XML	eXtensible Markup Language.

SUMÁRIO

1	INTRODUÇÃO	11
1.1	PROBLEMA	12
1.2	OBJETIVO GERAL	12
1.3	OBJETIVOS ESPECÍFICOS	12
1.4	JUSTIFICATIVA	12
2	REFERENCIAL TEÓRICO	14
2.1	Educação Infantil.....	14
2.2	Informática na Educação.....	15
2.3	<i>Software</i> de Referência.....	15
3	MATERIAIS E MÉTODOS	17
3.1	Metodologia de pesquisa	17
3.2	Metodologia de Desenvolvimento	18
4	RESULTADOS E DISCUSSÃO	21
4.1	Visão geral do software	21
4.2	Documento de requisitos.....	22
4.2.1	Escolher atividade	22
4.2.2	Realizar atividade	23
4.2.3	Avaliação do educador	23
4.2.4	Efetuar login	23
4.2.5	Realizar cadastro.....	24
4.3	Diagramas UML	24
4.3.1	Diagrama de caso de uso	24
4.3.2	Diagrama de classe	27
4.3.3	Diagrama Sequência	28
4.3.4	Diagrama de atividades	30
4.3.5	Diagrama de componentes.....	32
4.3.6	Diagrama de entidade e relacionamento	32
4.4	Objetivos Gerenciais do Projeto	32
4.4.1	Equipe de desenvolvimento.....	33
4.4.2	Sobre a metodologia.....	33
4.4.3	Product Backlog.....	34

4.5	Configurando ambiente de desenvolvimento	47
4.6	Desenvolvimento do aplicativo	48
4.6.1	Desenvolvimento da Interface do aplicativo	50
4.6.2	Comunicação e repositório	56
4.6.3	Testes.....	58
5	CONCLUSÃO	62
5.1	Trabalhos futuros	62
6	REFERÊNCIAS.....	64

1 INTRODUÇÃO

Com a tecnologia é possível desenvolver e realizar as mais diversas atividades, a utilidade desses recursos faz dos computadores e outros dispositivos um grande aliado, principalmente na educação, os computadores hoje são um fator de extrema importância na escola moderna.

ALMEIDA (1978) diz que a tecnologia pode criar uma curiosidade em buscar informações, dado e o prazer na escrita, ativar-lhes o desejo de enriquecer seu conhecimento seu diálogo.

TAJARA (2000) diz que as características a ser destacada na utilização da informática nas escolas é a interatividade que é proporcionada pelo computador, é a grande possibilidade que os computadores têm de ser uma ferramenta indispensável que facilitará o aprendizado (NASCIMENTO, 2009).

As TIC'S (tecnologia da comunicação e informação) podem ser utilizadas para o auxílio na aprendizagem trazendo o conhecimento de uma forma estruturada, estudar e usar as tecnologias da informação, "transformando o que é complicado em útil, prática e dinâmica além de ser mais criativo, e estimulante" (MARIA, 2010, p.128)

A utilização dos recursos que a tecnologia nos dispõe para favorecer as crianças, podem ser de extrema ajuda aquelas que tem dificuldade em aprender, pois pode indicar falhas tanto no ensino como na capacidade da criança aprender, assim desenvolvendo meios para que essas falhas sejam supridas e resolvendo os problemas que afetem a aprendizagem (MARIA, 2010)

Nesse trabalho, por meio da experiência de um dos autores em uma escola de ensino fundamental, percebeu-se que não possuía um *software* que auxiliasse o professor de maneira direta no quesito alfabetização, tendo isso em mente através de pesquisas de *softwares* e sites de educação infantil, foi desenvolvido um *software*, que busca melhorar e evitar alguns dos problemas existentes nos *softwares* e jogos educativos atuais, que são utilizados nos laboratórios de informática pelas escolas municipais.

Foi desenvolvido um *software* com foco no *feedback* para o professor, auxiliando mais diretamente a observação do progresso do aluno na alfabetização.

1.1 PROBLEMA

Dentre as diversas possibilidades de alfabetização de uma criança, a monitoria de um educador é fundamental para o sucesso dessa fase.

Contudo obter um feedback que auxilie a tomada de decisão de intervenção são problemas recorrentes para os profissionais desta área.

Diante essa inquietude, em que medida um software poderia proporcionar respostas rápidas e eficazes que contribuam com o planejamento do professor.

1.2 OBJETIVO GERAL

Desenvolver um software que auxilie os educadores no desenvolvimento educacional do aluno da educação infantil pelo sistema de feedback imediato, sendo desenvolvido na linguagem Python™.

1.3 OBJETIVOS ESPECÍFICOS

- Desenvolver um aplicativo educacional utilizando a linguagem Python™;
- Desenvolver um software com regras da Interface homem computador (IHC);
- Incrementar na proposta do aplicativo um *feedback* da progressão do aluno;
- Utilizar conceitos de exercício-e-prática e Demonstração no desenvolvimento;
- Modelar e projetar o software utilizando a linguagem de modelagem Unificada e metodologia ágil SCRUM;

1.4 JUSTIFICATIVA

Ao ter conhecimento de como funciona as aulas de alfabetização nos laboratórios de informática em uma escola pública E.M.E.I.E.F Professor Celso Augusto Rocco, percebemos que existe um limitado grupo de *softwares* educativos, desse grupo a maior parte dos *softwares* foram feitos apenas para Linux. Dos *softwares* existentes para Windows existem poucos exercícios para aprendizado do aluno, além do fato de os professores não terem a noção específica de como o aluno está aprendendo, pois esses *softwares* não dão um retorno específico sobre como o aluno está progredindo.

O nosso *software* busca trazer uma forma melhorada para auxílio da alfabetização dos alunos, com um *feedback* sobre o progresso do aluno e trazer uma alternativa para a falta de *software offline* no campo da alfabetização.

2 REFERENCIAL TEÓRICO

Será apresentado todo o recorte teórico que foi utilizado para alcançar resultado desejado, possui os conceitos e características do assunto abordado do ponto de vista de outros autores.

2.1 Educação Infantil.

As escolas vêm tentando se adequar com os alunos cada vez mais, diversos estudos e pesquisas são feitos para um melhor tipo de aprendizagem. O órgão brasileiro responsável pela área de alfabetização conduzido pela Secretaria de Alfabetização (Sealf) é a Política Nacional de alfabetização (PNA), como é descrito no Art. 1 do decreto nº 9.765.

Art. 1º Fica instituída a Política Nacional de Alfabetização, por meio da qual a União, em colaboração com os Estados, o Distrito Federal e os Municípios, implementará programas e ações voltados à promoção da alfabetização baseada em evidências científicas, com a finalidade de melhorar a qualidade da alfabetização no território nacional e de combater o analfabetismo absoluto e o analfabetismo funcional, no âmbito das diferentes etapas e modalidades da educação básica e da educação não formal. (Decreto nº 9.765, de 11 de abril de 2019, Brasil, 2019, pág. 15-16)

O ministério da educação (MEC) através do decreto nº 9204, de 23 de novembro de 2017, decreto nº 9.319 de 21 de março de 2018 na criação do programa educação conectada e junto com a comunidade acadêmica científica visa a implantação de sistemas de auxílio a educação básica utilizando a Tecnologias Digitais de Informação e Comunicação (TDIC). Que vem gerando resultados no ensino brasileiro. Na primeira etapa da educação básica as práticas pedagógicas têm como eixos a interação e as brincadeiras a ponto de garantir que as experiências possam ser realizadas utilizados dispositivos como gravadores, projetores e computadores.

De acordo com a base nacional comum curricular (BNCC) para educação infantil foi definido que na primeira etapa escolar existem 6 direitos de aprendizagem e desenvolvimento que são conviver, brincar, participar, explorar, expressar, conhecer-se, e os demais explorar movimentos, gestos, sons, formas, texturas, cores, palavras, emoções, transformações, Relacionamentos, Histórias, objetos, elementos

da natureza, na escola e fora dela, ampliando seus saberes sobre a cultura em suas diversas modalidades: as artes, a escrita, a ciência e a tecnologia (MEC, 2010).

2.2 Informática na Educação.

A história da utilização de computadores e *softwares* na educação brasileira começa de acordo com o livro Projeto Educom em 1971, onde foi discutido pelo governo, a utilização de computadores em uma universidade, sendo ela a USP de São Carlos (NASCIMENTO, 2009).

O objetivo dos computadores era o auxílio no ensino de física da universidade, entretanto registros indicam que a primeira universidade a se aplicar computadores em sua educação foi a Universidade Federal do Rio de Janeiro por meio do Núcleo de Cálculos Especiais (NCE).

Nessa época os computadores eram somente utilizados para pesquisa, em 1973 o Núcleo de Tecnologia Educacional para a Saúde (Nutes), Centro Latino-Americano de Tecnologia Educacional (Clates) da mesma universidade, começaram a se utilizar a informática no âmbito educacional formada para a educação e não somente pesquisa (NASCIMENTO, 2009).

Uma questão a ser abordada é quanto tempo cada atividade vai possuir, e de acordo com a Sociedade Brasileira de Pediatria (SBP) recomenda-se um tempo médio em que a criança deve ficar na frente de um computador ou uma tela em si, uma criança de 2 a 5 anos deve permanecer menos de 1 hora/dia já uma de 6 a 10 anos pode permanecer de 1 a 2 hora/dia e de 11 a 18 até 3 horas/dia.

A SPB também cita que as crianças precisam sempre de um responsável junto a elas. Observando essas recomendações vamos desenvolver um *software* com conjuntos de atividades que busca seguir cada uma delas (EISENSTEIN et al., 2019).

2.3 *Software* de Referência.

Um dos *softwares* educativos utilizado no Brasil é o Gcompris, um *software* francês para o auxílio a educação de crianças de 2 a 10 anos de idade. O *software* é de uso livre, e está em constante evolução, atualmente ele possui cerca de 100 atividades de diversas modalidades, atualmente o *software* não possui sistemas de

feedback ou um sistema que de um resultado ao professor da aprendizagem do aluno (GCOMPRIS, 2020).

O Code.org é uma instituição sem fins lucrativos, que visa buscar o acesso a ciência da computação, de modo as crianças terem acesso a todo o conteúdo de forma simples, de maneira que pode ser parte de sua educação no ensino fundamental e básico. O site é disponibilizado a pessoas de todo o mundo (MICROSOFT, 2021).

3 MATERIAIS E MÉTODOS

Será apresentado as metodologias científicas que vão ser utilizadas no desenvolvimento do *software*, assim como todos os materiais que foram utilizados nessa etapa do projeto.

3.1 Metodologia de pesquisa

A pesquisa vai ser desenvolvida nos parâmetros de estudo de caso, e das pesquisas exploratória, aplicada, documental e pesquisa de campo. A pesquisa exploratória é o método de pesquisa mais flexível e abrangente, ela trabalha na questão sobre se existe ou não um fenômeno e busca encontrar respostas a problemas que não são claros. Na pesquisa exploratória pode estar envolvido também a pesquisa bibliográfica ou entrevista com grupos, tornando melhor a necessidade do pesquisador para entender e testar a viabilidade de um estudo extenso ou escolher melhores métodos para um estudo. O objetivo dessa pesquisa é identificar questões-chave e variáveis-chave ¹.

A pesquisa aplicada é utilizada para resolver problemas de forma rápida e com as medidas necessárias. Ela busca resolver vários problemas utilizando os resultados da pesquisa fundamental. Nesse tipo de pesquisa o pesquisador coloca em prática os resultados da pesquisa fundamental. O foco dessa pesquisa são os fenômenos reais e questões científicas ².

A pesquisa documental conta com documentos analíticos e fontes de registro, como prontuários, testamentos, leis, revistas, jornais, catálogos, filmes, foto, entre outros (PRISCILA, 2019).

A pesquisa de campo é onde os pesquisadores entram diretamente no ambiente onde a pesquisa será realizada, e por meio dela podem observar, coletar,

¹ INTERNATIONAL NETWORK FOR NATURAL SCIENCES. Types of scientific research. [entre 2009 e 2020]. Disponível em: <https://innspub.net/types-of-scientific-research/>. Acesso em: 24 out. 2020.

² CANADIAN INSTITUTE FOR KNOWLEDGE DEVELOPMENT. Types of scientific research. 2019. Disponível em: <https://icndbm.com/types-of-scientific-research/>. Acesso em: 24 out. 2020.

analisar, e explicar fatos ou fenômenos. Por meio da pesquisa de campo o pesquisador adquire informações do que está estudando. Muito utilizada em áreas sociais, sendo dessa a sociologia e antropologia, nesse tipo de pesquisa podemos entender a cultura de um determinado grupo (PRISCILA, 2019).

3.2 Metodologia de Desenvolvimento

O *software* será desenvolvido nos parâmetros da orientação objeto, que consiste em um desenvolvimento em módulos independentes ou objetos e eles podem ser substituído, modificados e reutilizados, o *software* se torna uma coleção de objetos onde ficam encapsulado dados e operações, onde cada objeto é construído utilizando estruturas de dados (HENRIQUE, 2019).

O *Visual Studio Code* é um editor de código de forma simplificada, possui suporte para operações de desenvolvimento, sendo esses depuração, execução de tarefas e controle de versão. Seu objetivo é fornecer apenas as ferramentas de que um desenvolvedor precisa para um ciclo rápido de depuração de código e deixa fluxos de trabalho mais complexos para IDEs com recursos mais completos, como o IDE do Visual Studio. *VisualCode*© é gratuito para uso próprio e comercial³.

Não há hoje em dia critérios específicos da forma a se desenvolver um *software* educacional como o de auxílio a alfabetização, no entanto podemos utilizar a ISO/ESO 9126-1 que se refere a atributos para descrever e avaliar a qualidade de um *software* que sendo esses: Funcionalidade, confiabilidade, usabilidade e eficiência. Desenvolvendo assim o *software* nos pilares citados acima, para produzir um *software* com a maior qualidade.

O CorelDraw© é um dos *softwares* de *design* gráfico mais utilizados e possui uma vasta gama de ferramenta para *layout* e edição, e uma ferramenta de ponta que utiliza soluções de Inteligência Artificial para otimizar e auxiliar o usuário e assim sendo possível ser utilizado por todos os níveis de habilidade⁴.

A linguagem escolhida para a programação do *software* educativo foi Python™ na versão 3.7.9, pois é última versão estável lançada até momento. Python™ foi

³ VISUAL STUDIO CODE. **Visual Studio Code FAQ**. 2020. Disponível em: <https://code.visualstudio.com/docs/supporting/faq>. Acesso em: 30 nov. 2020.

⁴ CORELDRAW. **As empresas do mundo todo potencializam sua criatividade e sua produção com o CorelDRAW**. [201-]. Disponível em: <https://www.coreldraw.com/br/licensing/business/?topNav=br>. Acesso em: 30 nov. 2020.

escolhido por ser fácil e rápido de se programar, além de ser gratuito e de *software* aberto (*open source*). Um pouco de sua história, Python™ foi projetado por Guido Van Rossum em 1991 e foi desenvolvido pela *Python Software Foundation*. Com objetivo de permitir a programação com um dos conceitos de ter menos linhas de código. Guido Van Rossum começou trabalhar em Python™ em 1980, Guido ajudou na programação da linguagem ABC, porém, ele tinha muitas reclamações de problemas da linguagem, foi onde ele teve a ideia de construir uma nova linguagem melhorada (PRAMANICK, 2019).

Para interface gráfica (GUI - *Graphical User Interface*) iremos utilizar o PyQt5 e Qt designer. Nas buscas para uma interface gráfica para Windows o framework Qt foi escolhido por ser gratuito e fácil de implementar. Diferente da interface padrão do Python™ o Tkinter, o framework PyQt, de acordo com site Pypi possui diversas funcionalidades a mais do que o Tkinter, sendo algumas dessas o acesso a mais de 35 módulos, suporte a outras plataformas IOS e *Android* e sua implementação é baseada em C++, tornando mais rápida sua execução.

Qt é mais do que um kit de ferramentas GUI. Inclui abstrações de sockets de rede, threads, Unicode, expressões regulares, bancos de dados SQL, SVG, OpenGL, XML, um navegador da web totalmente funcional, um sistema de ajuda, uma estrutura de multimídia, bem como uma rica coleção de widgets GUI (**tradução nossa**)⁵.

Para gerar o executável, ou seja, o aplicativo que possa ser executado no Windows, iremos utilizar o FBS (*Fman Build System*) que é capaz de gerar um instalador para qualquer computador. O FBS utiliza Pyinstaller que é capaz de gerar um executável com todas dependências e bibliotecas necessárias para que o aplicativo possa funcionar sem a necessidade de possuir o Python™ instalado no computador (HERRMANN, 2016).

As principais vantagens do PyInstaller sobre ferramentas semelhantes são que o PyInstaller funciona com Python™ 3.5 a 3.7, PyInstaller cria executáveis menores graças à compressão transparente e é totalmente multiplataforma e usa o suporte do sistema operacional para carregar as

⁵ RIVERBANK COMPUTING. **What is PyQt?** [2016?]. Disponível em: <https://www.riverbankcomputing.com/software/pyqt/>. Acesso em: 30 set. 2020.

bibliotecas dinâmicas, garantindo assim total compatibilidade (**tradução nossa**)⁶.

Será utilizado um banco de dados offline chamado Qsqlite, o Qsqlite é junção do Qt com Sqlite. Sqlite é uma biblioteca que implementa o banco de dados SQL sem necessitar de um servidor ou de configurações, isto significa que ele é um servidor offline. O banco de dados é gerado na própria pasta da aplicação, como nossa aplicação será simples e totalmente offline esse será o banco de dados escolhido para ser implementado⁷.

Um dos processos no desenvolvimento de um *software* é a escolha de uma metodologia ágil, atualmente existem diversas metodologias e uma delas é a metodologia ágil Scrum, que foi desenvolvida por Jeff Sutherland e Ken Schwaber. O Scrum foi desenvolvido de forma que todo o processo e responsabilidade é dividido em três papéis o *Product owner*, Scrum master e o time, onde cada papel tem sua responsabilidade.

O Scrum trabalha com *Sprints*, que são reuniões diárias, semanais e mensais, onde em cada uma delas é trabalhado todo o conteúdo planejado para aquele período, onde cada um dos três papéis citados acima apresentam os resultados produzidos e os que estão em desenvolvimento.

Feedback: o *software* exibirá ao professor quantidade de acertos do aluno e as informações relacionadas de modo geral sobre o que o aluno fez.

Para o design será utilizada o *CorelDraw* em conjunto com banco de imagens de acesso gratuito na web.

⁶ PYINSTALLER. **PyInstaller**. 2020. Disponível em: <https://www.pyinstaller.org/>. Acesso em: 23 nov. 2020.

⁷ SQLITE. What Is SQLite?. [entre 2000 e 2020]. Disponível em: <https://www.sqlite.org/index.html>. Acesso em: 1 out. 2020.

4 RESULTADOS E DISCUSSÃO

Será documentado todo o material e os próprios resultados que foram obtidos durante o desenvolvimento do *software*.

4.1 Visão geral do software

O *software* de alfabetização tem como objetivo auxiliar na alfabetização de crianças e alunos. Ele vai ser desenvolvido para ser operado no sistema operacional Windows. O *software* vai contar com um sistema de *feedback*, para auxiliar os professores no andamento da alfabetização do aluno.

O *software* vai ser baseado em *softwares* já existente, como o Gcompris e Escola Games® em que o aluno irá realizar as atividades com textos e imagens.

Ele vai contar com inúmeras atividades de auxílio à alfabetização, o *software* vai começar com 03 atividades iniciais na primeira versão, posteriormente será acrescentado mais atividades.

As imagens que vão aparecer para o aluno terão seu formato em JPG e PNG. As caixas de texto vão ser mostradas em uma cor chamativa e centralizada para facilitar o aluno a encontrar e escrever, assim como as letras que o aluno vai selecionar vão ter uma espessura maior sendo também coloridas e chamativas assim facilitando o aluno a identificar e não deixando a atividade monótona.

Ao abrir *software* os alunos terão exercícios para serem feitos, antes disso o professor irá entrar com senha e nome do aluno para que ele possa observar o avanço do aluno.

Na primeira vez que for feito cadastro do aluno, o professor irá cadastrar para o aluno o nome do aluno e a senha assim como outros dados. Sendo possível mudar esses dados depois.

O *Software* usará os seguintes sistemas de funcionalidade *Point in click*: o usuário irá selecionar com *click* os exercícios, com diferentes Modos de se executar as atividades.

Um dos aspectos mais importante do projeto software é o *feedback* para o professor e por meio dele analisar o desempenho da criança que está sendo alfabetizada, o *feedback* será apresentado ao professor quando as atividades forem completadas ou quando o tempo acabar, o *feedback* apresentara ao professor quais atividades o aluno acertou e quanto tempo ele permaneceu na atividade, o professor terá duas opções para o *feedback*, a primeira forma ele apenas aparecerá no final de cada atividade e o professor poderá cadastrar um e-mail, e receberá os resultados dos seus alunos para que seja feita a análise deles.

Os dados que o professor encontrará em cada *feedback* serão: O número de acertos do aluno ou da criança e o tempo total que o aluno demoro para completar toda a atividade. No início de cada atividade o professor colocara o tempo recomendado para o aluno finalizar a atividade e no final da atividade o *feedback* vai ser enviado ao professor mostrando se o aluno completou a atividade no tempo sugerido pelo professor ou ele demorou mais para completar ou concluir em um intervalo de tempo menor. Assim auxiliando o professor na sua análise de como nivelar o aluno no *software*.

4.2 Documento de requisitos

Após a fase de levantamento de requisitos, foram identificados e coletados os seguintes requisitos para o software:

4.2.1 Escolher atividade

Tabela 1 - Documento de requisitos Escolher atividade

Requisito Funcional		
Nome: Escolher atividade		Código: F1
Descrição: Mostrar níveis de atividade pela série e idade do aluno		
Estimativa de Esforço: 3h		Prioridade: 100 pontos
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Básico da funcionalidade do sistema.	Usabilidade
1.2	As atividades dos alunos serão selecionadas de acordo como nível de conhecimento do aluno designado pelo professor.	Interface
1.3	O tempo máximo será 59 minutos para atividades.	Restrição

Fonte: Elaborado pelos autores (2020)

4.2.2 Realizar atividade

Tabela 2 - Documento de requisitos Realizar atividade

Requisito Funcional		
Nome:	Realizar atividade	Código: F2
Descrição:	Exibir imagens nos exercícios e exibir os exercícios com diferentes formas de resolução.	
Estimativa de Esforço:	20h	Prioridade: 100 pontos
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Não precisa de treinamento.	Usabilidade
1.2	A interface será de janela única.	Interface
1.3	As imagens serão no formato PNG.	Restrição

Fonte: Elaborado pelos autores (2020)

4.2.3 Avaliação do educador

Tabela 3 - Documento de requisitos Avaliação do educador

Requisito Funcional		
Nome:	Avaliação do educador	Código: F3
Descrição:	Mostrar feedback ao professor do total de acertos do aluno e tempo que aluno demorou para fazer as atividades propostas	
Estimativa de Esforço:	8h	Prioridade: 100 pontos
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Não precisa de treinamento.	Usabilidade
1.2	A interface será simples mostrando apenas o necessário sem muitas informações ou botões, sendo gerado uma imagem com os dados a mesma que será enviada por email.	Interface

Fonte: Elaborado pelos autores (2020)

4.2.4 Efetuar login

Tabela 4 - Documento de requisitos Efetuar Login

Requisito Funcional		
Nome:	Efetuar Login	Código: F4
Descrição:	Inserir o nome e a senha do aluno	
Estimativa de Esforço:	3h	Prioridade: 100 pontos
Requisitos Não funcionais		

ID NF	Descrição	Categoria
1.1	Não precisa de treinamento.	Usabilidade
1.2	A interface será de janela única.	Interface

Fonte: Elaborado pelos autores (2020)

4.2.5 Realizar cadastro

Tabela 5 - Documento de requisitos Realizar cadastro

Requisito Funcional		
Nome: Realizar cadastro		Código: F1
Descrição: Inserir nome, e-mail, senha, nome do professor		
Estimativa de Esforço: 6h		Prioridade: 100 pontos
Requisitos Não funcionais		
ID NF	Descrição	Categoria
1.1	Não precisa de treinamento.	Usabilidade
1.2	A interface será de janela única.	Interface
1.3	Para o cadastro será necessário nome, série e senha do aluno e o nome e email do professor.	Restrição

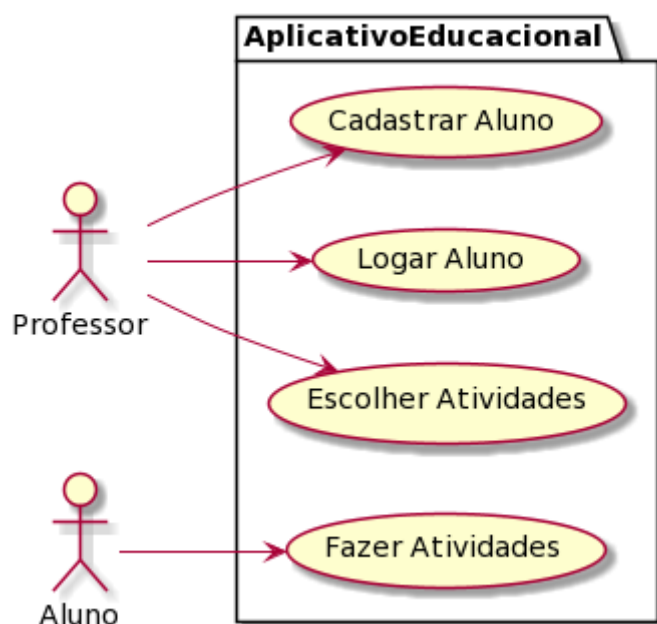
Fonte: Elaborado pelos autores (2020)

4.3 Diagramas UML

No processo de desenvolvimento de um *software* são produzidos diversos diagramas UML, será apresentado todos os diagramas que foram desenvolvidos no decorrer do processo de criação do *software*.

4.3.1 Diagrama de caso de uso

Figura 1 - Diagrama de casos de uso



Fonte: Elaborado pelos autores (2020)

4.3.1.1 Diagrama de caso de uso expandido

O diagrama de caso de uso será apresentado abaixo detalhadamente, contendo todos os eventos que ocorreram, para que se tenha uma visão ampla de todo o diagrama

4.3.1.1.1 Caso de uso – Cadastrar Aluno

Caso de Uso: Cadastrar Aluno (CSU01)

Descrição: O usuário deve cadastrar um aluno que ainda não consta na base de dados.

Ator Primário: Professor ou responsável.

Fluxo Principal

1. O usuário clica no botão Cadastro disponível na Tela Inicial.
2. O sistema abre a funcionalidade com todos os campos em branco para serem preenchidos
3. O sistema solicita dados obrigatórios, sendo eles Nome Completo do aluno, Idade, Nome do Professor, E-mail do professor e Senha.
4. O usuário preenche os dados solicitados.
5. O usuário clica no botão Cadastrar.

6. O sistema salva os dados no banco de dados.

Fluxo Alternativo 1: Excluir Aluno

1. O usuário pode desejar excluir o cadastro de um aluno.
2. O usuário faz login do Aluno.
3. O usuário clica no botão Excluir Aluno.
4. O sistema solicita confirmação da exclusão.
5. O usuário clica no botão Sim e faz a confirmação.

Fluxo Alternativo 2: Editar Aluno

1. O usuário pode desejar editar o aluno.
2. O usuário faz login do Aluno.
3. O usuário clica no botão Editar Aluno.
4. O sistema preenche os campos da funcionalidade com os dados do aluno.
5. O usuário clica no botão Editar.
6. O sistema solicita confirmação da edição.
7. O usuário clica no botão Sim e faz a confirmação.

4.3.1.1.2 Caso de uso – Logar Aluno

Caso de Uso: Logar Aluno (CSU02)

Descrição: O usuário deve logar um aluno que consta na base de dados.

Ator Primário: Professor ou responsável.

Precondições: O usuário deverá ter feito o cadastro do aluno

Fluxo Principal

1. Na Tela Inicial o usuário deverá clicar no botão Login.
2. Na tela de login o usuário deverá escolher o aluno entre os alunos carregados do banco de dados.
3. O usuário deverá colocar a senha do aluno.
4. O usuário clica no botão Login

4.3.1.1.3 Caso de uso – Escolher Atividades

Caso de Uso: Escolher Atividade (CSU03)

Descrição: o usuário deve escolher a atividades que aluno irá fazer.

Ator Primário: Professor ou responsável.

Precondições: O usuário deve ter feito o login do aluno

Fluxo Principal

1. Após ter feito login do aluno usuário será direcionado para tela de Escolha de atividades.
2. O usuário deverá escolher o tempo e qual atividade o aluno irá fazer.
3. O usuário deverá clicar no botão Fazer Atividade.

4.3.1.1.4 Caso de uso – Fazer Atividades

Caso de Uso: Fazer Atividades (CSU03)

Descrição: O Aluno deve fazer atividades escolhidas pelo responsável ou professor.

Ator Primário: Aluno

Atores Secundários: Professor ou responsável

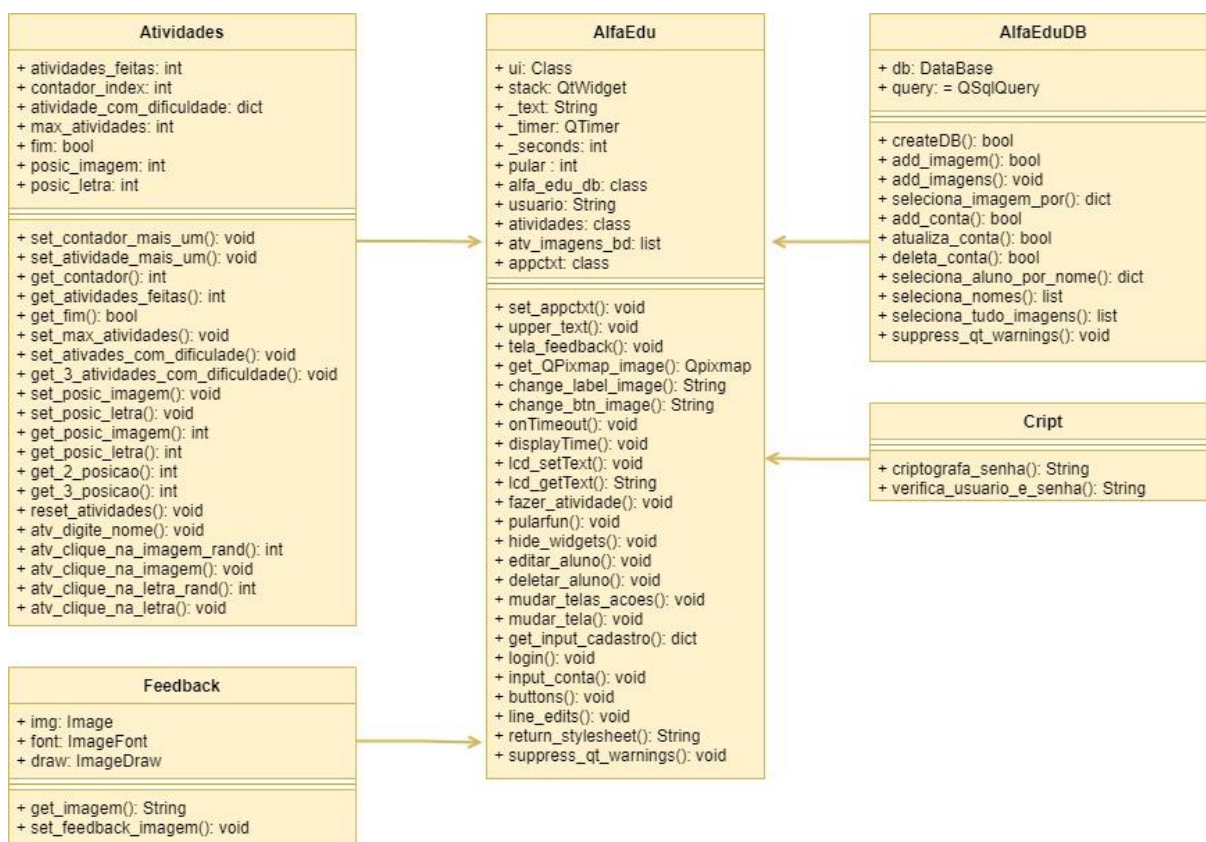
Precondições: As atividades devem ser escolhidas

Fluxo Principal

1. O aluno deverá responder as atividades na forma que aparecem cada atividade é diferente.

4.3.2 Diagrama de classe

Figura 2 - Diagrama de classe



Fonte: Elaborado pelos autores (2020)

4.3.3 Diagrama Sequência

O diagrama de sequência, representa a sequência de processos no *software*.

4.3.3.1 Cadastrar aluno

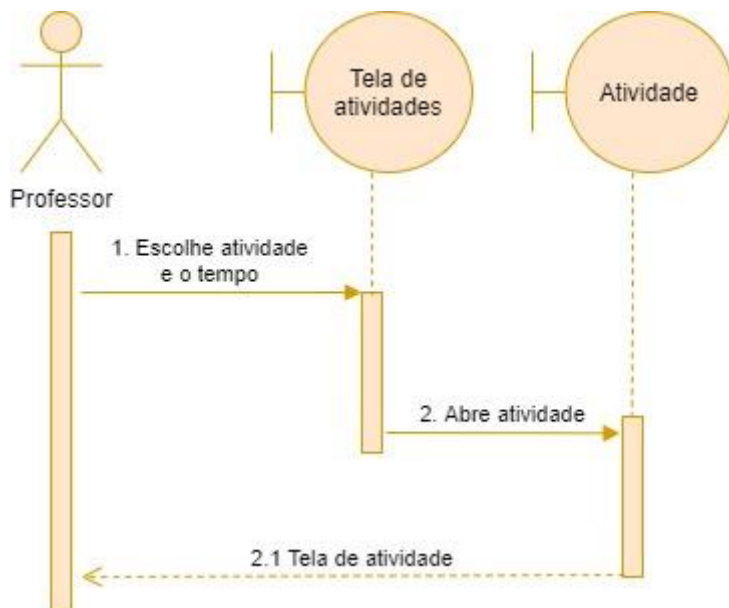
Figura 3 - Diagrama de sequência Cadastrar aluno



Fonte: Elaborado pelos autores (2020)

4.3.3.2 Escolher atividades

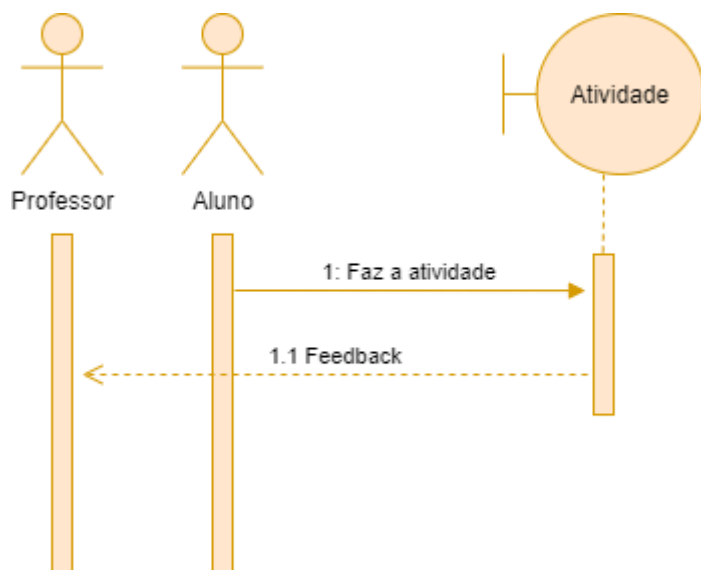
Figura 4 - Diagrama de sequência Escolher atividades



Fonte: Elaborado pelos autores (2020)

4.3.3.3 Fazer atividades

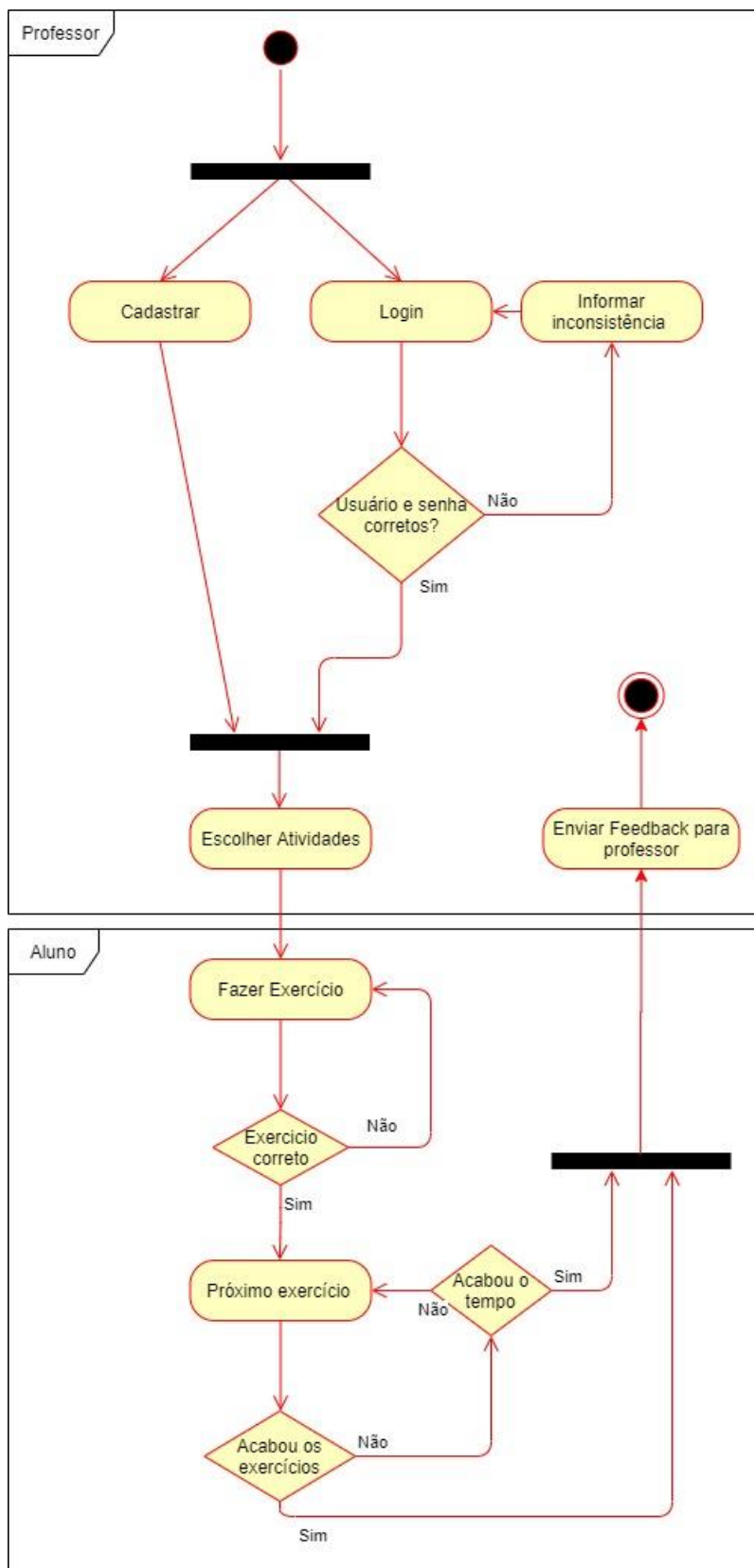
Figura 5 - Diagrama de sequência Fazer atividades



Fonte: Elaborado pelos autores (2020)

4.3.4 Diagrama de atividades

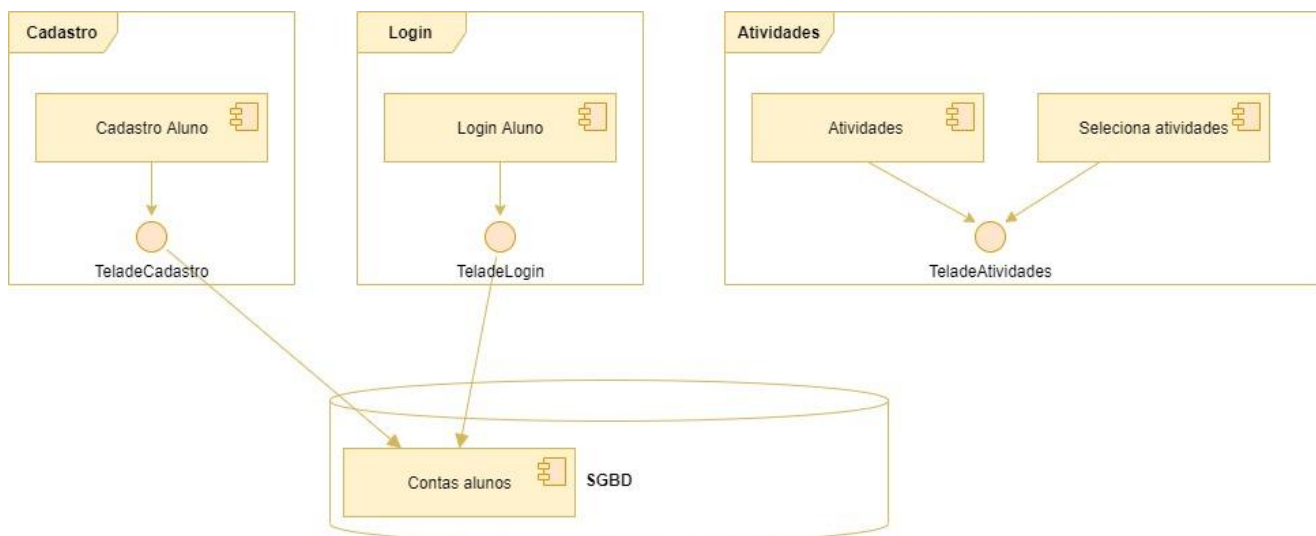
Figura 6 - Diagrama de atividades



Fonte: Elaborado pelos autores (2020)

4.3.5 Diagrama de componentes

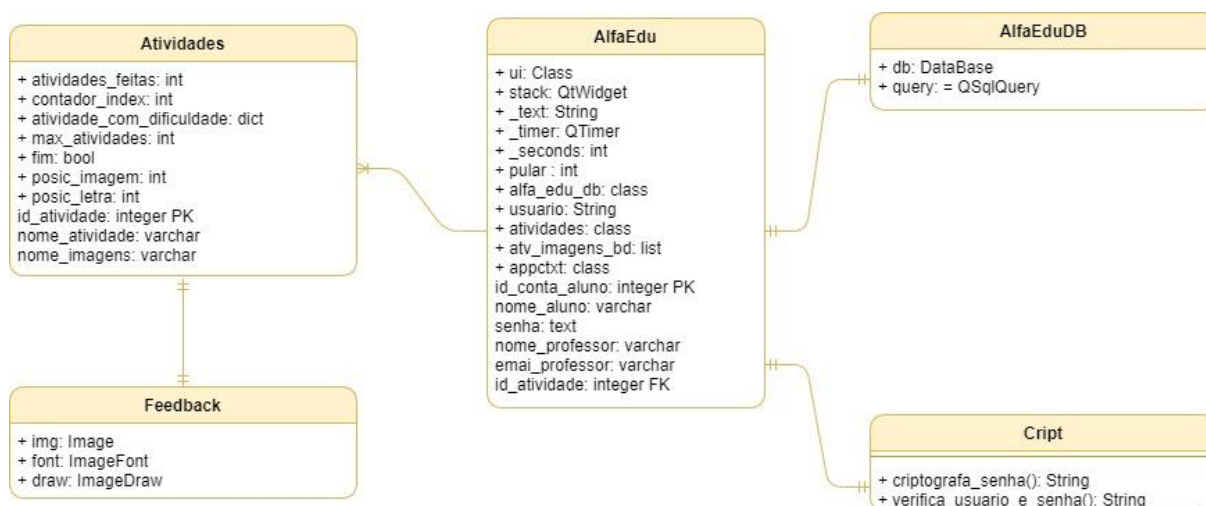
Figura 7 - Diagrama de componentes



Fonte: Elaborado pelos autores (2020)

4.3.6 Diagrama de entidade e relacionamento

Figura 8 - Diagrama de entidade e relacionamento



Fonte: Elaborado pelos autores (2020)

4.4 Objetivos Gerenciais do Projeto

Entregas: Durante o processo de desenvolvimento do software uma versão do produto deverá ser disponibilizada ao cliente ao final de cada sprint para que as funções desenvolvidas até determinado ponto sejam avaliadas pelo usuário.

Prazo: O prazo máximo para o desenvolvimento do software será de 12 (doze) meses;

Custos: Os custos previstos do projeto serão descritos nas Tabela 06 a seguir.

Tabela 6 - Custo Total do Projeto

Item	Descrição	Tempo Previsto	Valor Unt.	Valor Total
01	Valor Equipamentos para desenvolvimento	12 meses	R\$ 2.880,00 R\$ 8.000,00	R\$ 10.880,00
Total				R\$ 10.880,00

Fonte: Elaborado pelos autores (2021)

Manutenção: O aplicativo não terá custo de manutenção.

Contrato:

Comunicação: Será disponibilizado um Google Docs para a melhor comunicação, e vídeos pelo Youtube

4.4.1 Equipe de desenvolvimento

A equipe de desenvolvimento será composta pelos seguintes papéis e atribuições, de acordo com a metodologia **ágil Scrum** estabelecida para este projeto:

Tabela 7 - Equipe de desenvolvimento

Sprints	Nome	Função	Atribuições
01	Wellyngton de Jesus Arruda Junior	Scrum Master	Liderar a equipe.
	Profa. Ana Flávia	Product Owner	Ajudar no entendimento do negócio
	Wesley Rodrigues Guimarães	Time - Desenvolvedor	Desenvolver o projeto de software
	Wellyngton de Jesus Arruda Junior	Time - Desenvolvedor	Desenvolver o projeto de software

Fonte: Elaborado pelos autores (2021)

4.4.2 Sobre a metodologia

Uma das características fundamentais da metodologia Scrum, e que a torna tão potencialmente poderosa, é a ideia de repetição (ou iteração) e melhoria. Isso funciona em relação tanto ao produto a ser trabalhado, quanto a eficiência da própria equipe.

Primeiramente foi feito o Sprint Planning onde é definido e apresentado o Sprint backlog que sera demonstrado logo abaixo, o Scrum trabalha com Sprints foi decidido que os sprint fossem semanais acontecendo toda Segunda-feira e com reuniões mensais de entrega.

4.4.3 Product Backlog

Neste tópico serão documentados todos os itens do Product Backlog criados ao longo do desenvolvimento do projeto.

Tabela 8 - Product backlog

Item	Descrição	Estimativa de Esforço	Nível de Prioridade
1	Construção do Documento de Especificação de Software	50h	100
2	Construção do Documento de Requisitos de Software	50h	90
3	Criação do Diagramas UML de Casos de Uso, Atividade e componente.	30h	80
4	Inicia testes de compatibilidade entre as linguagens e softwares.	25h	90
5	Desenvolvimento da interface (Tela Inicial e Login).	40h	100
6	Desenvolvimento da interface(cadastro).	40h	100
7	Desenvolvimento do banco de Dados e script de enviar por e-mail.	55h	100
8	Desenvolvimento da interface das atividades e seu funcionamento.	150h	100
9	Adicionando criptografia ao software.	10h	100

10	Desenvolvimento do feedback.	20h	100

Fonte: Elaborado pelos autores (2021)

4.4.3.1 Sprints Backlog

Neste tópico serão documentados todos os itens dos Sprints Backlogs criados ao longo do desenvolvimento do projeto.

Tabela 9 - Sprints backlog

Item do Product Backlog	Nº Sprint e Período	Tarefa	Responsável	Data de Início	Previsão de Entrega
1	Sprint 1: 17/08/2020- 31/10/2020	Pesquisa e desenvolvimento referencial teórico	Wellyngton	17/08/2020	17/10/2020
		Pesquisa de linguagem e softwares.	Wesley	17/08/2020	17/10/2020
		Desenvolvimento dos materiais e métodos	Wellyngton e Wesley	01/10/2020	31/10/2020
2	Sprint 2: 27/10/2020 – 31/10/2020	Desenvolvimento dos requisitos de softwares	Wesley	27/10/2020	31/10/2020
3	Sprint 3: 27/10/2020- 10/11/2020	Construção dos diagramas: caso de uso, atividade e componente.	Wesley	27/10/2020	10/11/2020

4	Sprint 4: 11/10/2020- 05/01/2021	Início testes de compatibilidades da Qt designer.	Wesley	21/11/2020	05/01/2021
		Testes de compatibilidade com Fbs e início dos testes com banco de dados SQLite.	Wesley	11/10/2020	05/01/2021
		Finalização dos primeiros testes de compatibilidade.	Wesley	12/10/2020	05/01/2021
5	Sprint 5: 21/11/2020- 10/01/2021	inicia o desenvolvimento da tela inicial, assim como a criação do Main.ui e Main.py.	Wesley	21/11/2020	05/01/2021
		inicia o desenvolvimento do design da tela inicial	Wesley	05/01/2021	10/01/2021
		Centralizando tela inicial e removendo exemplos de testes	Wesley	06/01/2021	10/01/2021
		Diversas alterações na tela	Wesley	07/01/2021	10/01/2021

		inicial, como botão sair com Signal do QtDesigner, e adicionado Dialog para voltar para tela inicial da aplicação de atividades			
		Corrigindo bugs do Dialog	Wesley	08/01/2021	10/01/2021
6	Sprint 6: 10/01/2021- 17/01/2020	Criação da página de cadastro	Wesley	10/01/2021	17/01/2021
		Adicionado botão temporário que muda as páginas da aplicação para futuros testes.	Wesley	12/01/2021	17/01/2021
		Nova pasta de imagens criada e realocada todas as imagens para nova pasta.	Wesley	13/01/2021	17/01/2021
		Mudanças nas telas de cadastro e no nome das telas, foram adicionados comentários na programação.	Wesley	14/01/2021	17/01/2021

7	Sprint 7: 17/01/2021- 20/01/2020	Desenvolvido script de enviar e-mail, que será utilizado no feedback.	Wesley	17/01/202 1	20/01/202 1
		Adicionado banco de dados da aplicação para usuários.	Wesley	18/01/202 1	20/01/202 1
		Diversas alterações no código e no banco de dados.	Wesley	19/01/202 1	20/01/202 1
7,9	Sprint 8- 20/01/20201 - 01/02/2021	adição das imagens das atividades e telas.	Wellyngton	20/01/202 1	01/02/202 1
		criação da primeira tela de teste.	Wellyngton	27/01/202 1	01/02/202 1
		corrigiu um erro no import ocasionado pela adição de Imagem.	Wesley	27/01/202 1	28/01/202 1
		Foi iniciado o desenvolvimento da criptografia de senha do aluno	Wesley	20/01/202 1	01/02/202 1
		Foi Iniciado os testes unitários	Wesley	21/01/202 1	01/02/202 1

		para banco de dados			
		Foi feito Downgrade do Python para versão 3.6 para evitar problemas futuros de compatibilidades com FBS.	Wesley	22/01/2021 1	01/02/2021 1
		Iniciado o desenvolvimento das telas de atividades.	Wesley, Wellyngton	26/01/2021 1	01/02/2021 1
6	Sprint 9: 02/02/2021- 11/02/2021	Recria a tela inicial utilizando o método de design com frame.	Wellyngton	02/02/2021 1	11/02/2021 1
		Recriando a tela cadastro utilizando os frames.	Wellyngton	03/02/2021 1	11/02/2021 1
		Finalizada a tela cadastro começando a desenvolver o design da primeira atividade.	Wellyngton	04/02/2021 1	11/02/2021 1
		Finalizada a primeira atividade	Wellyngton	04/02/2021 1	11/02/2021 1

		iniciando a segunda.			
6,5	Sprint 10: 11/02/2021- 23/02/2021	Pequenas alterações no layout.	Wellyngton	11/02/2021	23/02/2021
		Adicionando contador de atividades e mais alterações no layout.	Wellyngton	16/02/2021	23/02/2021
		Adicionando fonte ao projeto.	Wellyngton	22/02/2021	23/02/2021
7	Sprint 11: 23/02/2021- 10/03/2021	Foram feitas diversas alterações no gerenciamento do banco de dados, como adição dos nomes das imagens	Wesley	23/02/2021	24/02/2021
		Alterou cabeçalho do aplicativo	Wellyngton	23/02/2021	24/02/2021
		Foi corrigido o bug do ApplicationContext que travava o aplicativo, mais nomes de imagens adicionadas no banco de dados.	Wesley	24/02/2021	25/02/2021

		Continuando desenvolvimento da atividade de digitar o nome da figura	Wesley	25/02/2021 1	26/02/2021 1
		Iniciado desenvolvimento da tela de feedback	Wesley	26/02/2021 1	10/03/2021 1
8	Sprint 12: 10/03/2021- 19/03/2021	Diversas alterações na atividade de digitar o nome da figura, agora figura troca apenas quando está certa a resposta	Wesley	10/03/2021 1	19/03/2021 1
		Resolvido bug de index que tentava acessar imagem com index acima da lista	Wesley	13/03/2021 1	19/03/2021 1
		Foi alterado a classe atividades para nova atividade de clicar na figura correta	Wesley	15/03/2021 1	19/03/2021 1
	Sprint 13: 19/03/2021- 25/03/2021	Diversas alterações na atividade de clicar na figura, alterando a	Wesley	19/03/2021 1	25/03/2021 1

8		exibição das imagens			
		Finalizando a atividade de clicar na figura	Wesley	22/03/2021	25/03/2021
		Reorganizando código da classe main e atividades, iniciado o desenvolvimento da tela editar aluno, função de adicionar e remover aluno iniciadas.	Wesley	24/03/2021	25/03/2021
		Adicionado a tela da atividade clique na letra que falta, editado a tela de escolher as atividades.	Wellyngton	24/03/2021	25/03/2021
8	Sprint 14: 25/03/2021- 31/03/2021	Finalizando a função de adicionar e remover aluno	Wesley	25/03/2021	31/03/2021
		Terminada atividade clique na letra que falta	Wesley	26/03/2021	31/03/2021
		Adicionado novas imagens no banco de dados	Wesley	31/03/2021	31/03/2021

8	Sprint 15: 01/04/2021- 26/04/2021	Adicionado novas imagens no aplicativo	Wellyngton	01/04/202 1	02/04/202 1
		Diversas alterações no layout do aplicativo	Wellyngton	02/04/202 1	03/04/202 1
		Alterações no feedback.	Wesley	01/04/202 1	26/04/202 1
8	Sprint 16: 26/04/2021- 26/05/2021	Adicionar imagem de fundo, recriar botões no Corel Draw, correção de bugs e teste finais.	Wesley, Wellyngton	26/04/202 1	26/05/202 1

Fonte: Elaborado pelos autores (2021)

4.4.3.2 Informações sobre as Sprints

Tabela 10 - Informações sobre as sprints

Sprint	Reunião	Resumo da Reunião	Data da Reunião
1	Planejamento do projeto, Pesquisa.	O Wellyngton iniciou a reunião, com a sugestão de software a ser desenvolvido pelo grupo. Foi passado a cada participante o objeto de pesquisa de cada um.	17/08/2020
2	Pesquisa.	Foi apresentado pelos membros do projeto a pesquisa feita na semana, Wellyngton pediu ao Wesley foco na pesquisa de fermentas. E para o Wellyngton foi decidido a pesquisa mais aprofundada nas normas do MEC sobre a alfabetização infantil.	24/9/2020

3	Desenvolvimento.	Foi finalizada a pesquisa, e desenvolvido o referencial teórico, na reunião foi apresentado pelo Wesley o desenvolvimento dos requisitos de software e que iniciaria a construção dos diagramas: sendo eles de caso de uso, atividade e componente.	31/10/2020
4	Testes e finalização.	Wesley inicia a reunião dizendo que irá começar os testes no QtDesigner, FBS, SQLite, e apresenta todos os diagramas construídos enquanto o Wellyngton irá finalizar o desenvolvimento dos materiais e métodos do artigo a ser entregue.	07/11/2020
5	Entrega	Foi feito os últimos ajustes para a entrega a banca da primeira parte, o artigo.	10/11/2020
6	Primeira reunião 2021, Desenvolvimento	Wesley Iniciou desenvolvimento da tela inicial do aplicativo, e apresentou os testes feitos durante esse intervalo sem comunicação. Wellyngton comunicou que iria iniciar a pesquisa sobre Design da interface.	05/01/2021
7	Desenvolvimento da tela	Wesley apresenta as telas iniciais desenvolvidas, Wellyngton questiona a organização, e diz que encontrou uma solução para melhorar o design, mas ainda está estudando o funcionamento e a aplicação.	13/01/2021
8	Banco de dados, Imagens	Wesley diz que o banco de dados está em desenvolvimento, e que o script de enviar por e-mail já está implementado,	20/01/2021

		teve alterações no banco que já estava pronto. Wellyngton apresenta as novas imagens que foram adicionadas no banco de dado das atividades.	
9	Downgrade, Desenvolvimento, atividades.	Wesley inicia avisando que foi feito o Downgrade na versão do Python utilizada, a versão a ser utilizada foi a 3.6, foi preciso realizar o Downgrade para evitar futuros problemas com FBS. Wesley diz que iniciou o desenvolvimento das atividades, e pediu a wellyngton que acelerasse o desenvolvimento do layout. Wellyngton disse que ainda está trabalhando no layout, e finalizando os testes.	27/01/2021
10	Telas em frames.	Wellyngton inicia dizendo que finalizou os testes, e diz que vai recriar as telas utilizando se do método de frames. Wesley continua no desenvolvimento backend.	05/02/2021
11	Informações.	Wesley e Wellyngton avisa um ao outro, que estão desenvolvendo as mesmas atividades do sprint anteriores.	13/02/2021
12	Layout, Banco de dados.	Wellyngton ainda está desenvolvendo o layout, foi adicionado o contador de atividades entre outras alterações como no cabeçalho, adicionado uma fonte ao projeto. Wesley foram feitas diversas alterações no gerenciamento do banco de dados, como adição dos nomes das imagens	23/02/2021

13	Bugs, Feedback.	Wesley inicia a reuniao dizendo que encontrou bugs na atividade, mas já conseguiu resolver e que adicionou todas as imagens no banco de dados. Diz que vai continuando desenvolvimento da atividade de digitar o nome da figura, e que já começou o desenvolvimento do feedback.	02/02/2021
14	Desenvolvimento.	Não teve ouve nenhum reuniao e cada um estava trabalhando em sua determinada parte, ouve pequenas trocas de mensagens, dizendo o andamento, se estavam prestes a terminar ou não. E a cada 2 dias perguntávamos um ao outro quanto tempo levaria.	09/02/2021- 15/03/2021
15	Bugs e Imagens.	Wesley iniciou mostrando as mudanças que ele fez nas atividades, e como está ficando os exercícios, relatou que encontrou bugs, mas eles já foram solucionados. Wellyngton, está buscando imagens, e confeccionado o logo do app.	15/03/2021
15	Back-end atividades.	Wellyngton inicia, dizendo que finalizou as imagens, e que está enviando ao banco de dados, e que está quase terminando a última atividade. Wesley diz que está terminando de desenvolver a as atividades do back-end.	24/03/2021
16	Script, tela de atividades.	Wesley inicia a reuniao mostrando as alterações feitas no codigo main, e	01/04/2021

		começou a desenvolver o script para editar e remover aluno. Wellyngton adicionou a tela que faltava da atividade. E ambos adicionaram imagens ao banco de dados.	
16	Finalização do aplicativo.	No período de 01/04/2021 a 26/04/2021 não houve reunião. No dia 26/04 o Wellyngton inicia a reunião mostrando as imagens e o novo layout do software, como sua nova fonte botões. O Wesley apresenta o feedback e parte do código que foram modificados. Nessa reunião comprovam que o software está concluído e fica a cargo do Wesley os testes finais e correções de bug. Havendo necessidade de alteração a comunicação será feita pelo aplicativo Discord sem precisar de uma reunião. Assim ficou marcado a última reunião para dia 26/05/2021.	26/04/2021

Fonte: Elaborado pelos autores (2021)

4.5 Configurando ambiente de desenvolvimento

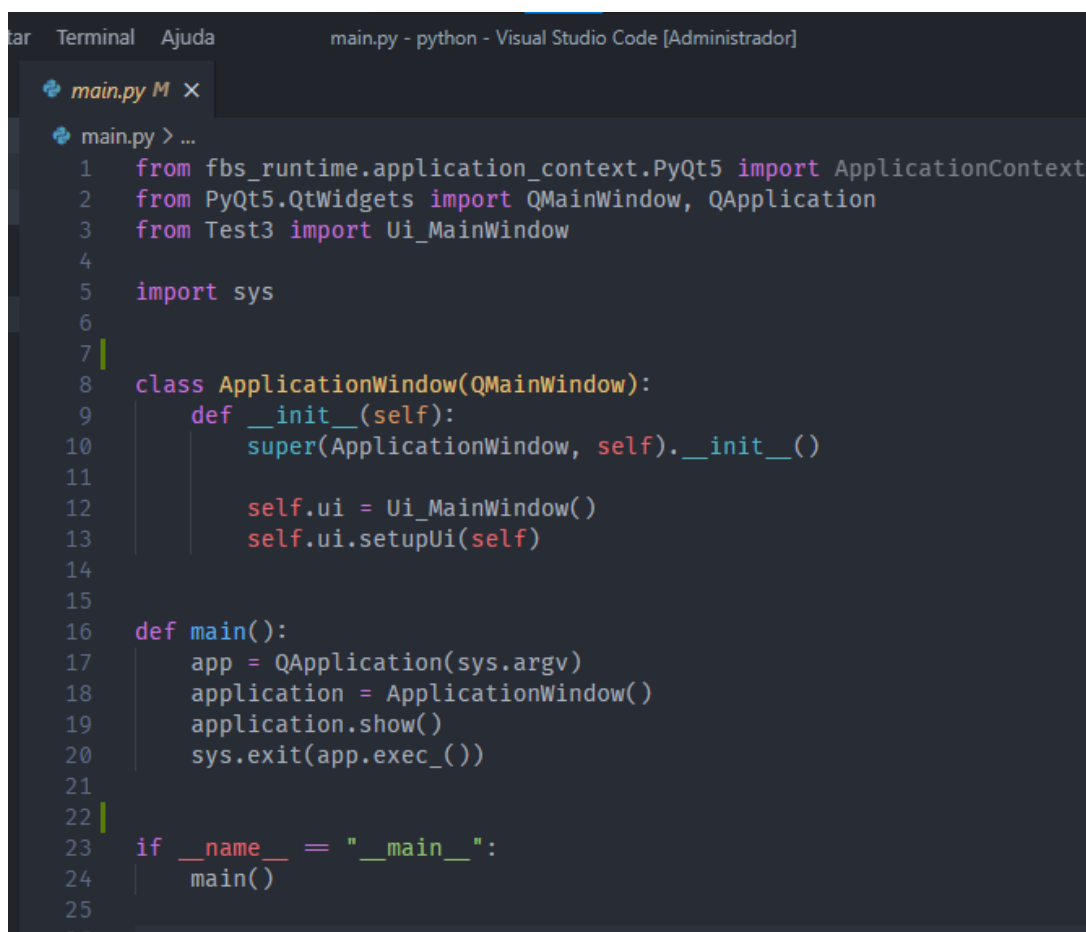
Antes de inicializar o desenvolvimento do *software*, foi configurado e instalado o ambiente de desenvolvimento e os aplicativos e *plugins* necessários. A primeira parte foi instalar o Python™ na versão 3.6.7 que foi escolhida para ser a linguagem de programação para o app, após foi instalado e configurado o Visual Studio Code, seguindo de algumas bibliotecas que foram instaladas pelo PIP (Sistema de gerenciamento de pacotes do Python™), a primeira biblioteca foi o FBS (*Fmain Build System*) que é a biblioteca responsável por criar o projeto Python™ com a conexão com QT por meio do comando `fbs startproject nome do projeto`, depois foi criada a interface do Qt, .ui e convertida para python .py

utilizando o seguinte comando “python -m PyQt5.uic.pyuic -x mainUi.ui -o mainUi.py” , para ter funcionamento melhor no aplicativo. Depois foi configurado o ambiente no Git e Github® para ser desenvolvido em outros computadores.

4.6 Desenvolvimento do aplicativo

Com o ambiente instalado e configurado foi iniciado o desenvolvimento do *software*, inicialmente foi instalado as bibliotecas necessárias, bcrypt que era biblioteca responsável pela criptografia da senha do usuário, porém não foi utilizada pois não estava funcionando com o projeto. Abaixo figura 9 estão as primeiras linhas de código que foram digitadas. Ela gera a primeira tela do app, que foi utilizada para testar se todas as bibliotecas que foram utilizadas eram compatíveis.

Figura 9 - Código da primeira tela criada



```

1  from fbs_runtime.application_context.PyQt5 import ApplicationContext
2  from PyQt5.QtWidgets import QMainWindow, QApplication
3  from Test3 import Ui_MainWindow
4
5  import sys
6
7
8  class ApplicationWindow(QMainWindow):
9      def __init__(self):
10         super(ApplicationWindow, self).__init__()
11
12         self.ui = Ui_MainWindow()
13         self.ui.setupUi(self)
14
15
16  def main():
17     app = QApplication(sys.argv)
18     application = ApplicationWindow()
19     application.show()
20     sys.exit(app.exec_())
21
22
23  if __name__ == "__main__":
24     main()
25
26

```

Fonte: Elaborado pelos autores (2021)

Como apresentado na figura 9, após os testes foi criado a tela inicial, de cadastro e login, onde toda a interface foi projetada e desenvolvida no Qt e adicionado as funcionalidades pelo arquivo main.py.

Figura 10 - Código da imagem de feedback

```
teste1.py > ...  
    from PIL import Image, ImageFont, ImageDraw  
  
    img = Image.new("RGBA", (1280, 720), "white")  
    titulo = "Feedback"  
  
    font = ImageFont.truetype("arial.ttf", 50)  
    largura, altura = font.getsize(titulo)  
  
    draw = ImageDraw.Draw(img)  
    # draw.text(((1280-largura)/2, (720-altura)/2), titulo  
    draw.text((100, 2), titulo, font=font, fill="black")  
    img.save("testando.bmp")  
    # img.show()
```

Fonte: Elaborado pelos autores (2021)

A figura 10 mostra o código da criação da imagem de feedback, a biblioteca utilizada para gerar essa imagem foi a Pillow, essa biblioteca é utilizada para processamento de imagem para Python™, e no nosso projeto ela foi utilizada para transformar textos em imagem, assim pegando todas as informações que o software gera durante a sessão de atividades do aluno. E com isso gerar uma imagem que será primeiramente apresentada no final da atividade, e que em seguida será enviada por e-mail. Para realizar o envio do e-mail foi utilizada uma biblioteca chamada SMTPLIB que tem a característica de ser um protocolo de e-mail, que é utilizado para enviar o feedback direto ao endereço do professor. Como vemos na figura 11 o código em que essa biblioteca é adicionada ao software.

Figura 11 - Código de envio do e-mail

```

3 import smtplib
4 from email.mime.multipart import MIMEMultipart
5 from email.mime.text import MIMEText
6 from email.mime.base import MIMEBase
7 from email import encoders
8
9 # Configuração
10 # conexão com os servidores do google
11 host = 'smtp.gmail.com'
12 port = 465
13
14 # username ou email para login no servidor
15 user = 'exemplo@gmail.com'
16 # TODO criptografar senha
17
18 # TODO Acessar BD
19 password = 'exemplosenha'
20
21 try:
22     # Criando objeto
23     print('Criando objeto servidor ... ')
24     # conectando de forma segura usando SSL
25     server = smtplib.SMTP_SSL(host, port)
26
27     # server.ehlo()
28     # server.starttls()
29
30     # Login com servidor
31     print('Login ... ')
32     server.login(user, password)

```

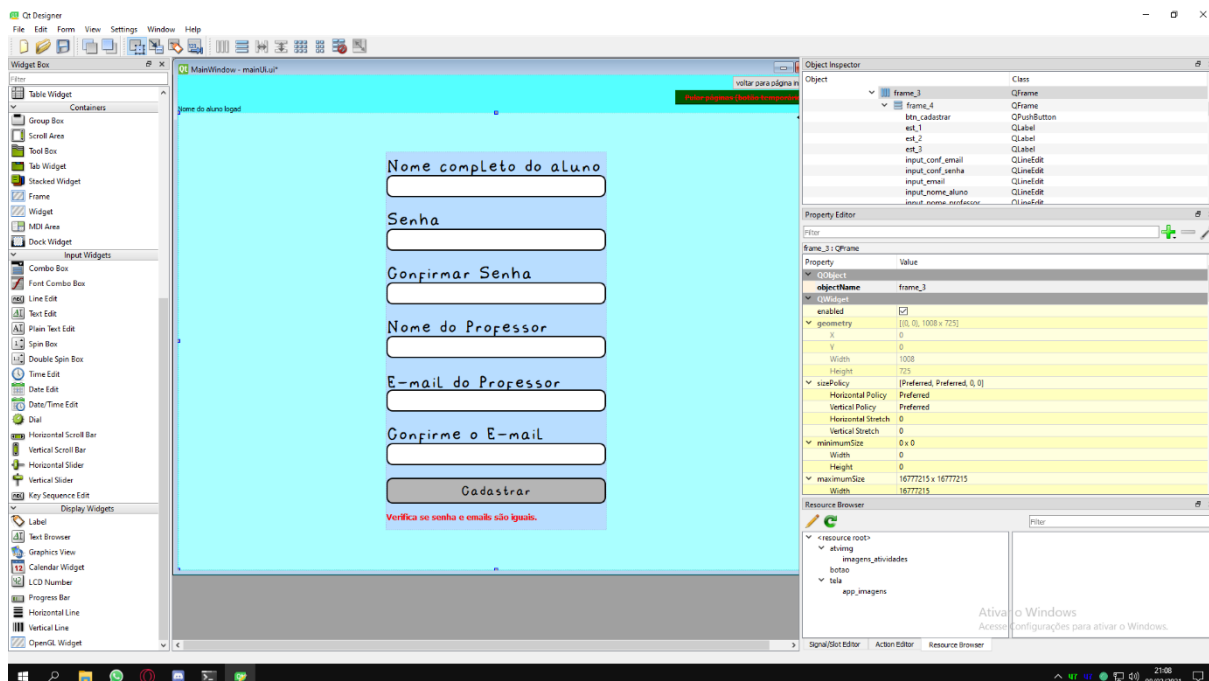
Fonte: Elaborado pelos autores (2021)

Observação: os dados de email utilizado no print não são o mesmo do produto final.

4.6.1 Desenvolvimento da Interface do aplicativo

Como apresentado anteriormente a ferramenta para se desenvolver toda a parte de interface foi o QT designer. Toda interface foi criada utilizando o método de frames. Que são molduras que facilitam o desenvolvimento além de transformar a interface em responsiva de uma maneira mais eficiente e fácil.

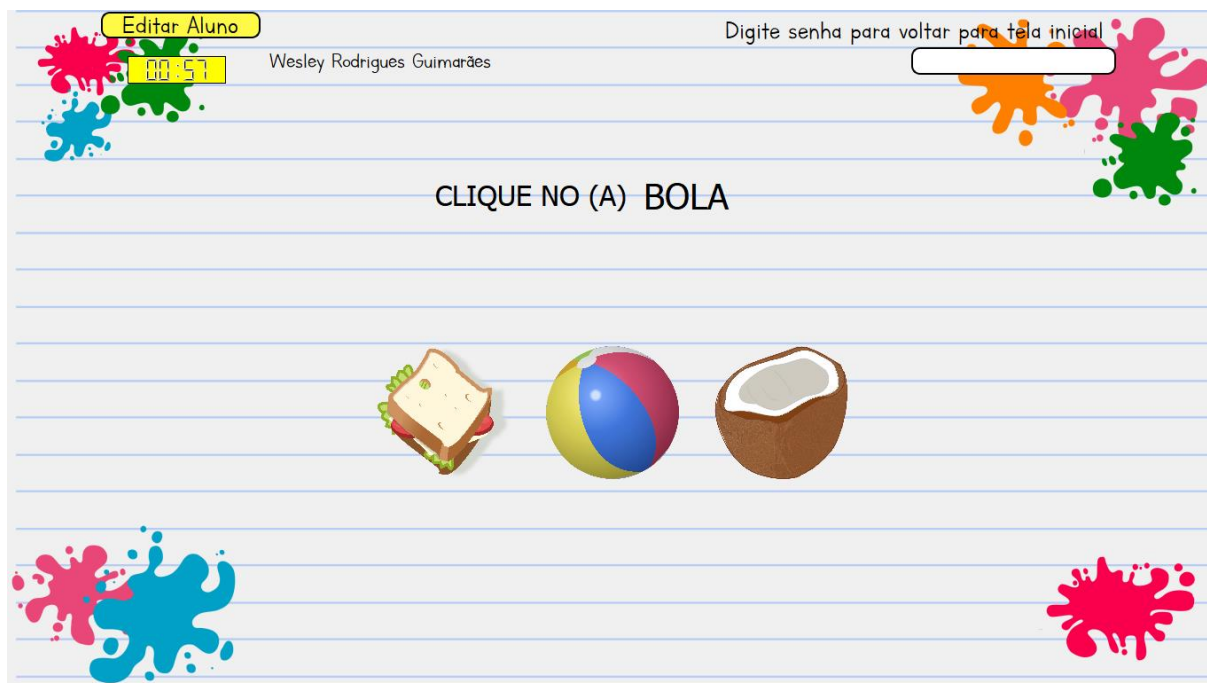
Figura 12 - Tela de cadastro



Fonte: Elaborado pelos autores (2021)

Na figura 12 pode-se observar o desenvolvimento da interface de cadastro no QT, utilizando os frames. Toda a interface foi projetada de maneira a deixar o aplicativo com o menor tamanho possível e com um designer amigável ao usuário que sendo estas as crianças. Terminada toda as telas de login e cadastro, foi iniciado o desenvolvimento das atividades, elas foram feitas com base em atividades impressa existentes nas escolas. Iniciamos com três atividades.

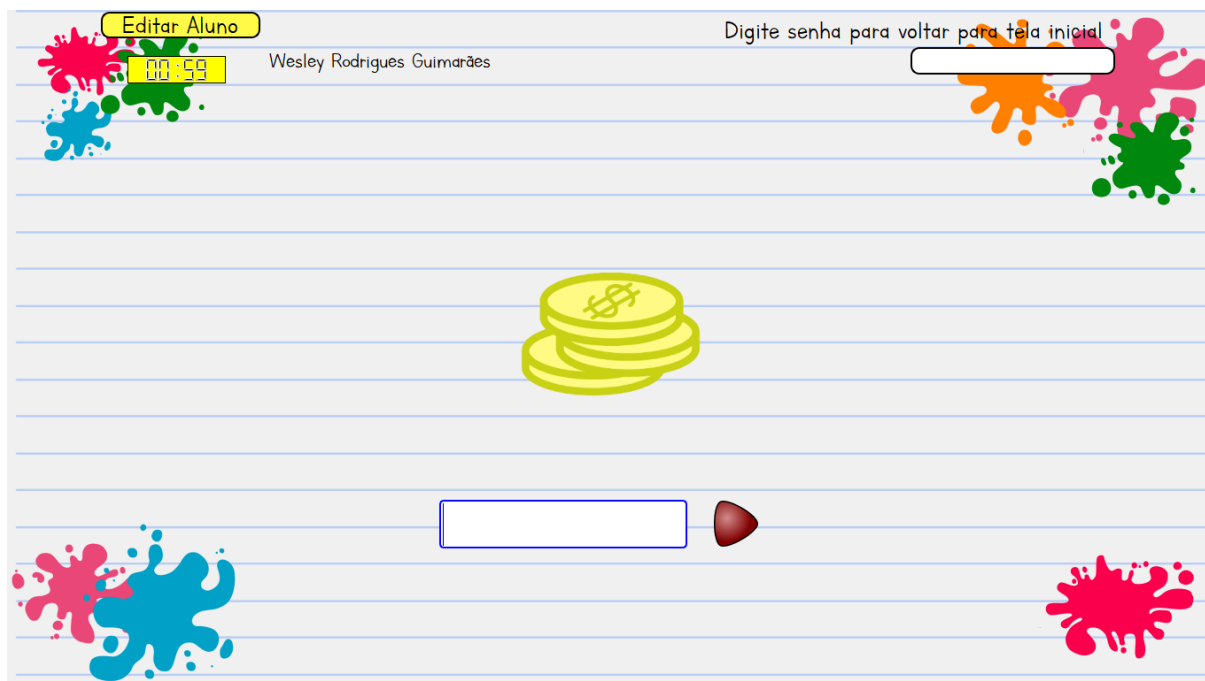
Figura 13: Atividade clique na figura



Fonte: Elaborado pelos autores (2021)

A primeira atividade a ser desenvolvida foi a clique na figura como o nome já sugere, o aluno vai clicar na imagem correspondente ao texto. As imagens da atividade estão na pasta principal do app. Na atividade as respostas dos alunos são salvas pelo script de atividades para posteriormente ser apresentada ao feedback. Ainda percorrendo sobre imagens, há em todas as atividades um script para que os exercícios apareçam de forma aleatória.

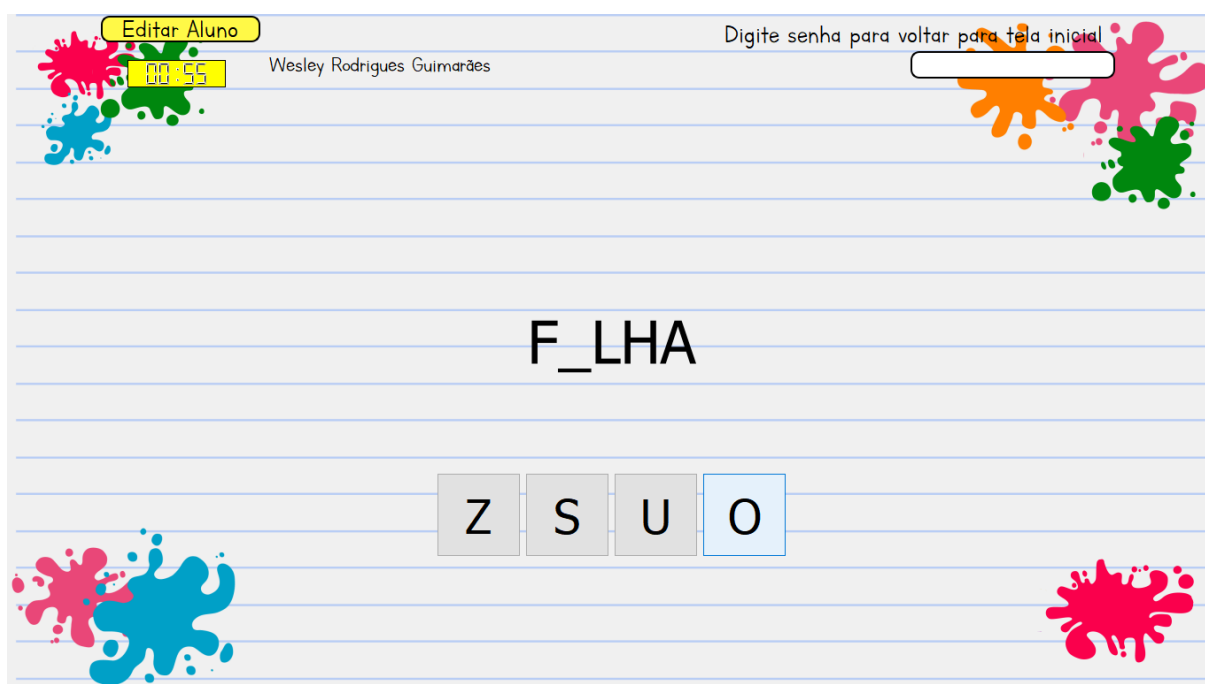
Figura 14 - Atividade digite o nome da figura



Fonte: Elaborado pelos autores (2021)

A próxima atividade figura 14 funciona da seguinte forma, aparecerá ao aluno uma imagem e uma caixa de texto, o aluno deverá escrever o nome da imagem, assim como na primeira, caso o aluno digite o nome incorretamente, esse dado será salvo para que posteriormente apareça na página de feedback.

Figura 15 - Atividade clique na letra que falta

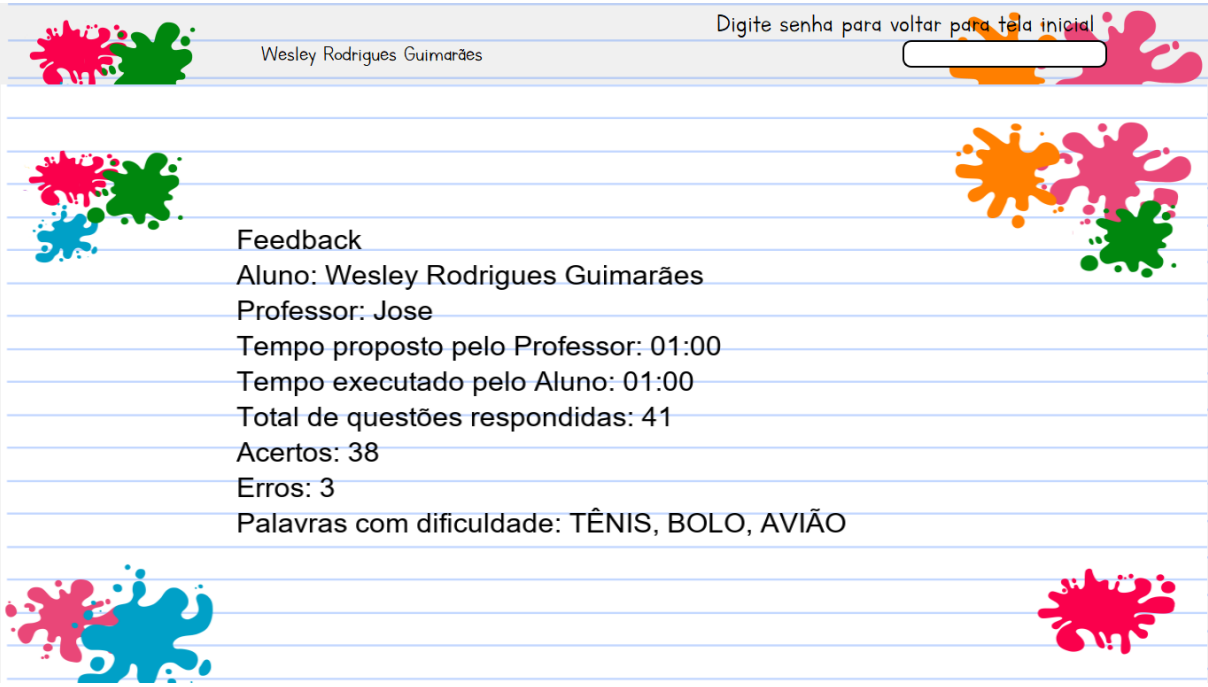


Fonte: Elaborado pelos autores (2021)

A terceira e última atividade figura 15 desenvolvida nesse período funciona da seguinte forma, na tela será apresentado ao aluno uma palavra faltando uma letra, e logo abaixo vão aparecer letras, e o objetivo é o aluno selecionar a caixa onde está a letra correspondente a que falta na palavra acima.

Abaixo nas figuras 16,17,18,19 temos as imagens das telas de feedback, login, cadastro e editar aluno. Que não foram apresentadas acima.

Figura 16 - Tela de feedback

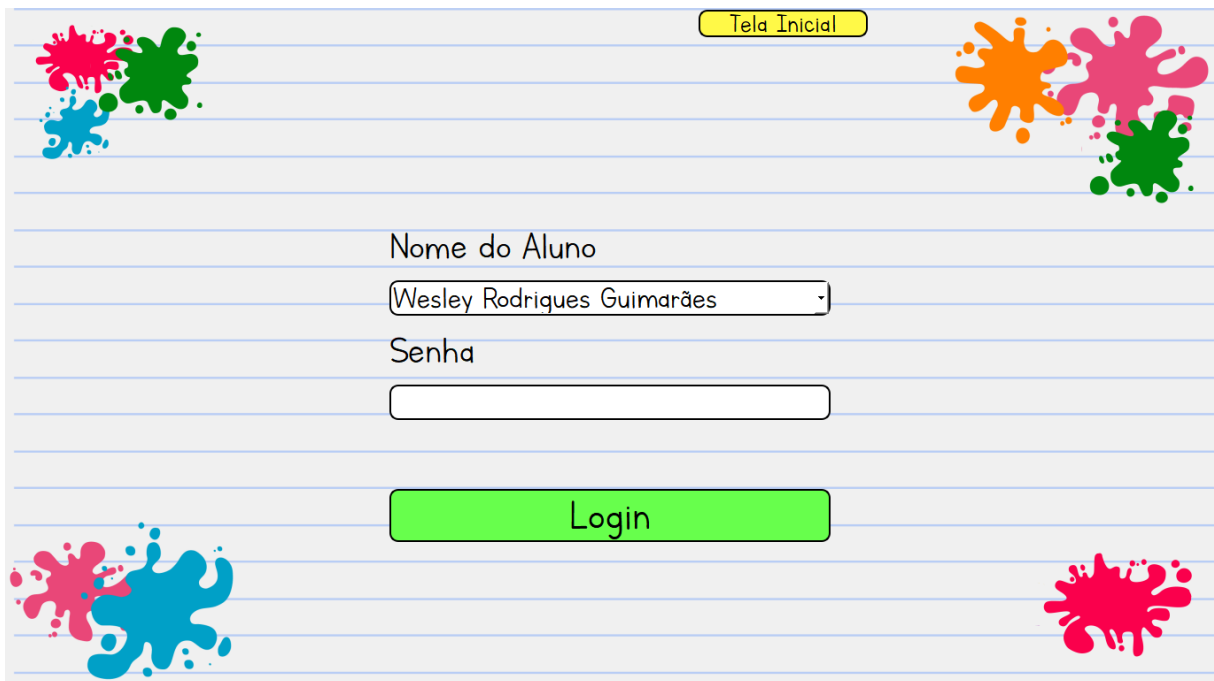


The screenshot shows a feedback screen with a header bar containing the student's name 'Wesley Rodrigues Guimarães' and a button labeled 'Digite senha para voltar para tela inicial'. The main content area displays the following feedback information:

- Feedback
- Aluno: Wesley Rodrigues Guimarães
- Professor: Jose
- Tempo proposto pelo Professor: 01:00
- Tempo executado pelo Aluno: 01:00
- Total de questões respondidas: 41
- Acertos: 38
- Erros: 3
- Palavras com dificuldade: TÊNIS, BOLO, AVIÃO

Fonte: Elaborado pelos autores (2021)

Figura 17 - Tela de login

A interface de login apresenta um fundo cinza com linhas horizontais azuis. No topo, há uma barra amarela com o texto "Tela Inicial". As bordas são decoradas com manchas coloridas (verde, amarelo, laranja, rosa, azul). O formulário centralizado contém o rótulo "Nome do Aluno" acima de um campo de texto com o nome "Wesley Rodrigues Guimarães" e uma seta de seleção. Abaixo dele está o rótulo "Senha" e um campo de senha. No final, há um botão verde com o texto "Login".

Tela Inicial

Nome do Aluno

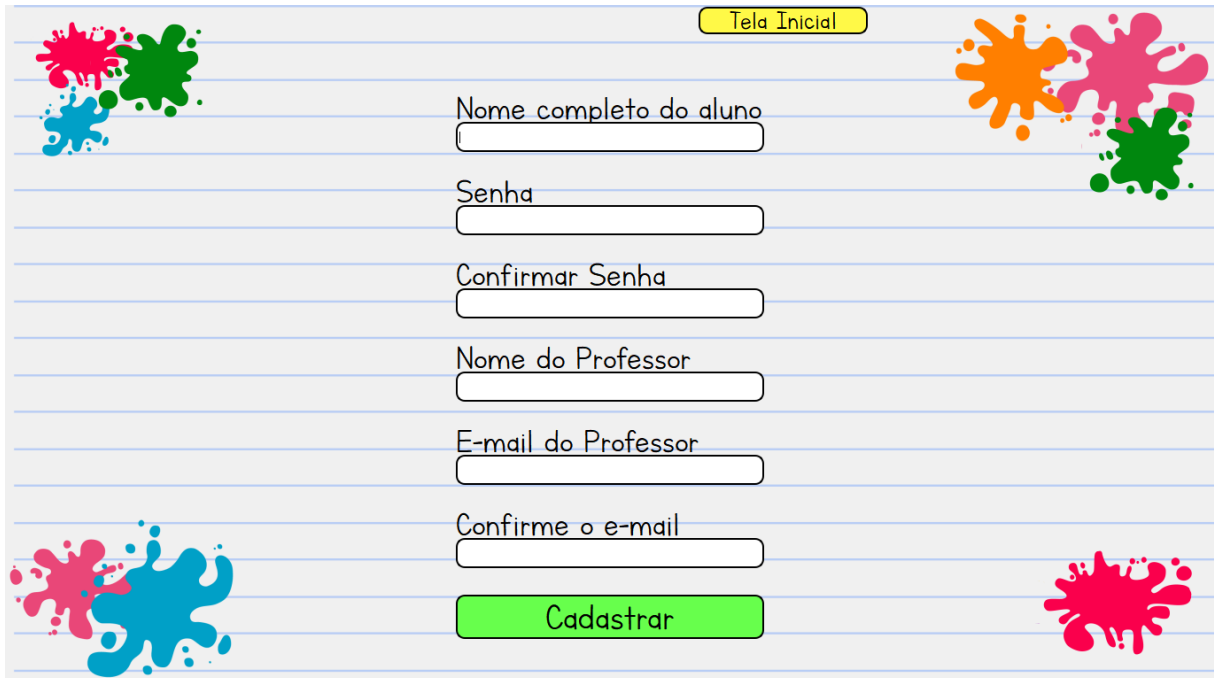
Wesley Rodrigues Guimarães

Senha

Login

Fonte: Elaborado pelos autores (2021)

Figura 18 - Tela de cadastro

A interface de cadastro possui o mesmo design visual quanto a tela de login, com fundo cinza, linhas azuis e decorações coloridas. No topo, também há uma barra amarela com o texto "Tela Inicial". O formulário centralizado inclui os seguintes campos: "Nome completo do aluno" (campo de texto), "Senha" (campo de texto), "Confirmar Senha" (campo de texto), "Nome do Professor" (campo de texto), "E-mail do Professor" (campo de texto), e "Confirme o e-mail" (campo de texto). No final, há um botão verde com o texto "Cadastrar".

Tela Inicial

Nome completo do aluno

Senha

Confirmar Senha

Nome do Professor

E-mail do Professor

Confirme o e-mail

Cadastrar

Fonte: Elaborado pelos autores (2021)

Figura 19 - Tela de editar dados do aluno

Editar Aluno Tela Inicial

Wesley Rodrigues Guimarães

Nome completo do aluno
Wesley Rodrigues Guimarães

Senha
.....

Confirmar Senha
.....

Nome do Professor
Jose

E-mail do Professor
wesleyrg2000@gmail.com

Confirme o e-mail
wesleyrg2000@gmail.com

Editar

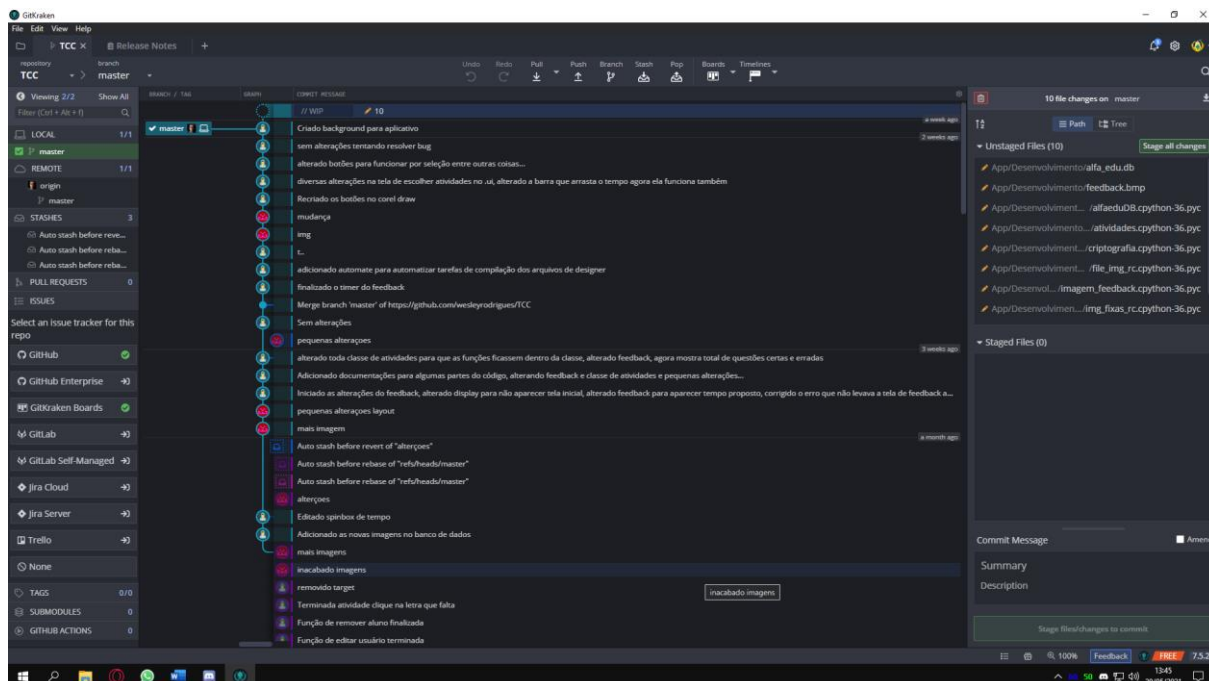
Excluir

Fonte: Elaborado pelos autores (2021)

4.6.2 Comunicação e repositório

Utilizamos como repositório de código o GitHub®, abaixo uma imagem dos Push que foram feitos para o repositório em nuvem. O aplicativo apresentado abaixo é o GitKraken que é utilizado para se observar todo o repositório em uma única interface.

Figura 20 - GitKrakem

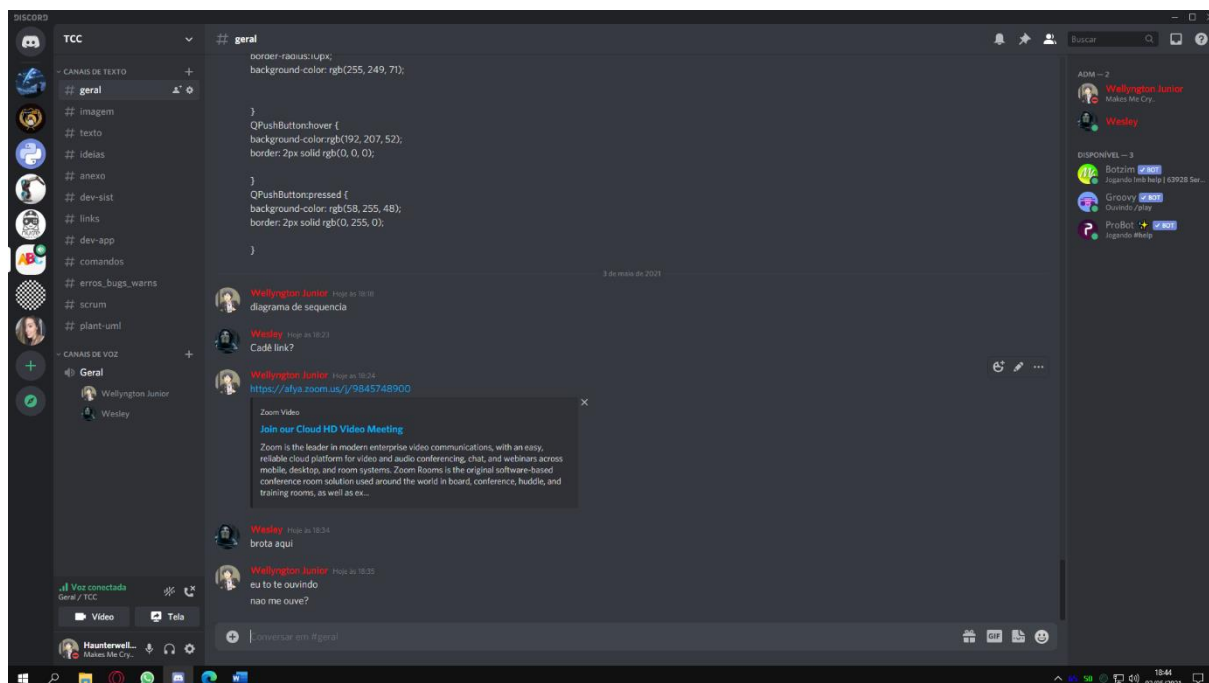


Fonte: Elaborado pelos autores (2021)

Como apresentado acima, para um melhor controle das modificações feitas no aplicativo cada passo do desenvolvimento foi nomeado de acordo com a modificação feita.

Devido a pandemia do COVID-19 e as restrições que foram impostas a sociedade, todas nossas reuniões foram em ambiente virtual de maneira remota, para essas reuniões utilizamos o software chamado Discord, que é um software de voz sobre IP, ele é gratuito e disponível em Windows, Linux, Mac OS, Android e IOS, além de ser um aplicativo de voz em grupo ele possui chat de texto, que foi utilizado por nós como repositório de links, mídias e demais informações. Abaixo apresentamos uma imagem, da plataforma em uma de nossas reuniões.

Figura 21 - Discord

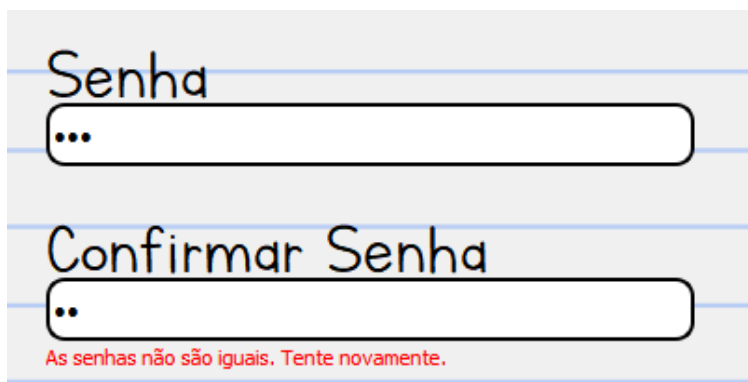


Fonte: Elaborado pelos autores (2021)

4.6.3 Testes

Foram realizados diversos testes para comprovar o funcionamento do programa, utilizamos para realizar os testes o FBS pois é possível observar se as ações feitas estão sendo salvas e processadas corretamente. Primeiramente foi realizado os testes no cadastro, umas das funções empregadas no cadastro é a conferir se as senhas são iguais e se os e-mails são iguais verificamos que está funcionando corretamente como apresentado na figura 22 e figura 23. após analisar essas funções completamos todo o cadastro e enviamos, como apresentado na figura 24 o aluno cadastrado foi salvo no banco de dados.

Figura 22 - Confirmação de senha



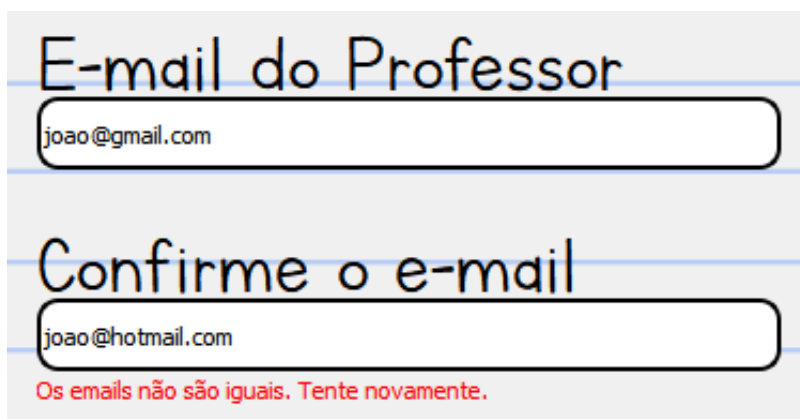
Senha

Confirmar Senha

As senhas não são iguais. Tente novamente.

Fonte: Elaborado pelos autores (2021)

Figura 23 - Confirmação de e-mail



E-mail do Professor

Confirme o e-mail

Os emails não são iguais. Tente novamente.

Fonte: Elaborado pelos autores (2021)

Figura 24 - Teste inserção do aluno no banco de dados

```

PowerShell
PowerShell 7.1.0
Copyright (c) Microsoft Corporation.

https://aka.ms/powershell
Type 'help' to get help.

A new PowerShell stable release is available: v7.1.3
Upgrade now, or check out the release page at:
https://aka.ms/PowerShell-Release?tag=v7.1.3

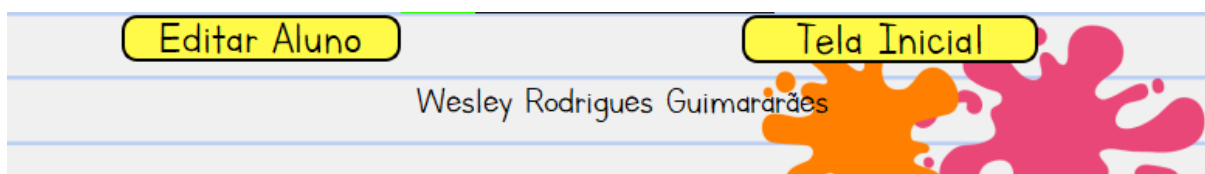
PS D:\TCC\App\Desenvolvimento> fbs run
libpng warning: iCCP: known incorrect sRGB profile

Resultado adicionado = True
  
```

Fonte: Elaborado pelos autores (2021)

Depois que foi executado o teste no cadastro passamos para o login, onde apareceu corretamente o nome do aluno e ao digitar a senha tivemos sucesso e foi efetuado o login. Na tela onde se encontra para selecionar as atividades tivemos êxito em mostrar o nome do Aluno como o botão de editar as informações do aluno logado demonstrado na figura 25, aproveitamos e testamos a parte de editar aluno ao entrar na tela efetuamos uma mudança e foi verificado que essa alteração foi salva no banco de dados.

Figura 25 - Botão de editar aluno



Fonte: Elaborado pelos autores (2021)

Prosseguido o teste vamos para a parte da atividade, foram testadas todas as atividades e o funcionamento delas foi o esperado, o tempo para se realizar a atividade, e as questões apareceram corretamente. Ao final da atividade recebemos

o feedback, observamos que todos os dados do feedback contam corretamente, uma observação e que não foi programado para que o feedback fique salvo no banco de dados, porém como ele é enviado por e-mail, verificamos que ele fica salvo na caixa de entrada do endereço de e-mail do professor, tivemos uma dificuldade no quesito envio, percebemos que o Gmail.com trava o acesso do aplicativo ao endereço de e-mail. O que pode gerar um erro que é resolvido ao permitir o aparelho no próprio site do Gmail.com, a forma com que o feedback chega ao e-mail do professor é representado na figura 25.

Figura 26 - Feedback no email



Fonte: Elaborado pelos autores (2021)

5 CONCLUSÃO

Com o passar do desenvolvimento desse trabalho de conclusão de curso, observamos as adversidades que se encontra em desenvolver um software e suas implicações passamos por diversas dificuldades, mas no fim é uma alegria ver que conseguimos desenvolver um software somente nosso, percebemos quão complexo e a fase de alfabetização de uma criança assim como a falta de tecnologias que auxiliem os professores alfabetizadores.

Esse trabalho teve o objetivo de produzir um software que auxilie os professores na alfabetização utilizando a linguagem Python e com um feedback direto ao professor.

Com a utilização do Python e suas bibliotecas fomos capazes de desenvolver o software apresentado, com todas as funcionalidades propostas.

Junto com o QTdesigner foi desenvolvida uma interface com características simples pois o software foi projetado para crianças na alfabetização.

Utilizando as regras de interface homem computador para dar ao usuário a melhor experiência ao interagir com o software.

Junto com o software foi desenvolvido um sistema de feedback para auxiliar o professor a verificar o progresso do aluno, os exercícios foram baseados no exercício de exercício-e-prática que são utilizados nas escolas.

A equipe utilizou a modelagem Unificada e metodologia ágil SCRUM, fazendo os sprint semanais para discussão sobre o desenvolvimento.

5.1 Trabalhos futuros

Após o desenvolvimento do software começamos a buscar e pensar como podemos estar melhorando o programa que acabamos de desenvolver, buscamos informações em fóruns de discussão, que possuem pessoas da área e decidimos quais são os próximos passos que vão ser implementados no software AlfaEdu.

- O desenvolvimento de novas atividades, atividades que abrange uma gama maior de conteúdo, pois a alfabetização tem várias etapas e vamos buscar meios para atingir a maioria dessas etapas.

- Armazenar o feedback no banco de dados, percebemos que o feedback armazenado somente no e-mail pode gerar confusão ao usuário. Então posteriormente vai ser adicionado uma forma de visualizar eles direto do programa, onde eles estarão salvo no banco de dados.
- Adicionar mais conteúdo visual, o conteúdo visual presente atualmente não engloba todos os tipos de palavras que a língua portuguesa possui.

6 REFERÊNCIAS

BRASIL. Decreto nº 9.765, de 11 de abril de 2019. Institui a Política Nacional de Alfabetização. Brasil: Diário oficial da união, p. 15-16, 11 abr. 2019.

CANADIAN INSTITUTE FOR KNOWLEDGE DEVELOPMENT. **Types of scientific research**. 2019. Disponível em: <https://icndbm.com/types-of-scientific-research/>. Acesso em: 24 out. 2020.

CORELDRAW. **As empresas do mundo todo potencializam sua criatividade e sua produção com o CorelDRAW**. [201-]. Disponível em: <https://www.coreldraw.com/br/licensing/business/?topNav=br>. Acesso em: 30 nov. 2020.

EISENSTEIN, Evelyn; PFEIFFER, Luci; GAMA, Marco Chaves; ESTEFENON, Susana; CAVALCANTI, Suzy Santana. **Manual de Orientação: #MENOS TELAS #MAIS SAÚDE**. SBP, 2019. 11 p. Disponível em: https://www.sbp.com.br/fileadmin/user_upload/_22246c-ManOrient_-_MenosTelas__MaisSaude.pdf. Acesso em: 15 set. 2020.

GCOMPRIS. **GCompris**. 2020. Disponível em: https://gcompris.net/index-pt_BR.html. Acesso em: 27 set. 2020.

HENRIQUE, João. **POO: o que é programação orientada a objetos?** 2019. Disponível em: <https://www.alura.com.br/artigos/poo-programacao-orientada-a-objetos>. Acesso em: 16 nov. 2020.

HERRMANN, MICHAEL. **Python and Qt: simplified!**. 2016. Disponível em: <https://build-system.fman.io/>. Acesso em: 1 out. 2020.

INTERNATIONAL NETWORK FOR NATURAL SCIENCES. **Types of scientific research**. [entre 2009 e 2020]. Disponível em: <https://innspub.net/types-of-scientific-research/>. Acesso em: 24 out. 2020.

MARIA, Isabel; VIRGÍLIA Luciana, **O Uso da Tecnologia Como Facilitadora da Aprendizagem do Aluno na Escola**. Itabaiana: Gepiadde, v. 8, 2010. Disponível em: <http://atividadeparaeducacaoespecial.com/wp-content/uploads/2014/08/USO-DA-TECNOLGIA.pdf>. Acesso em: 15 maio 2021.

MEC. **A educação infantil na base nacional comum curricular**. 2010. Disponível em: <http://basenacionalcomum.mec.gov.br/abase/#infantil>. Acesso em: 15 set. 2020.

MESH, Janet. **Método Kanban: Guia detalhado e 5 modelos prontos para usar**, 24 mar. 2020. Disponível em: <https://blog.trello.com/br/metodo-kanban>. Acesso em: 29 set. 2020.

MICROSOFT (org.). **Sobre nós**. 2021. Disponível em: <https://code.org/about>. Acesso em: 03 maio 2021.

NASCIMENTO, João Kerginaldo Firmino do. **Informática aplicada à educação**. Brasília, 2009. 84 p. ISBN 978-85-230-0981-6.

PRAMANICK, Sohom. **History of Python**. GeeksforGeeks, 6 maio 2019. Disponível em: <https://www.geeksforgeeks.org/history-of-python/>. Acesso em: 5 nov. 2020.

PRISCILA. **Conheça os principais tipos de pesquisas científicas para trabalhos acadêmicos**. 2019. Disponível em: <https://www.tuacarreira.com/tipos-de-pesquisa/>. Acesso em: 24 out. 2020.

PYINSTALLER. **PyInstaller**. 2020. Disponível em: <https://www.pyinstaller.org/>. Acesso em: 23 nov. 2020.

RIVERBANK COMPUTING. **What is PyQt?** [2016?]. Disponível em: <https://www.riverbankcomputing.com/software/pyqt/>. Acesso em: 30 set. 2020.

SQLITE. **What Is SQLite?** [entre 2000 e 2020]. Disponível em: <https://www.sqlite.org/index.html>. Acesso em: 1 out. 2020.

VISUAL STUDIO CODE. **Visual Studio Code FAQ**. 2020. Disponível em: <https://code.visualstudio.com/docs/supporting/faq>. Acesso em: 30 nov. 2020.