

**FAI – CENTRO DE ENSINO SUPERIOR EM GESTÃO,
TECNOLOGIA E EDUCAÇÃO
CURSO DE SISTEMAS DE INFORMAÇÃO**

CARLOS CHRISTIAN ZANIN

FELIPE LEME SIMÕES BALESTRA

GABRIEL RODRIGO MARTINS DA SILVA

**EDUQ STUDENT LIFE: SISTEMA DE SOFTWARE DE APOIO AO
ACOMPANHAMENTO PEDAGÓGICO NO ENSINO
FUNDAMENTAL – SÉRIES INICIAIS**

SANTA RITA DO SAPUCAÍ - MG

2019

**FAI – CENTRO DE ENSINO SUPERIOR EM GESTÃO,
TECNOLOGIA E EDUCAÇÃO
CURSO DE SISTEMAS DE INFORMAÇÃO**

CARLOS CHRISTIAN ZANIN

FELIPE LEME SIMÕES BALESTRA

GABRIEL RODRIGO MARTINS DA SILVA

**EDUQ STUDENT LIFE: SISTEMA DE SOFTWARE DE APOIO AO
ACOMPANHAMENTO PEDAGÓGICO NO ENSINO
FUNDAMENTAL – SÉRIES INICIAIS**

Projeto de final de curso apresentado à FAI – Centro de Ensino Superior em Gestão, Tecnologia e Educação, como requisito parcial para obtenção do Título de Bacharel em Sistemas de Informação, sob a orientação da profa. Eunice Gomes de Siqueira.

SANTA RITA DO SAPUCAÍ - MG

2019

FOLHA DE APROVAÇÃO

[Na impressão final do documento, esta folha será substituída pela assinada pela Banca examinadora final.]

HISTÓRICO DE REVISÕES

Data	Versão	Autor(es)	Descrição
10/12/19	4.8	Carlos, Felipe e Gabriel	Versão corrigida.
24/11/19	4.7	Carlos, Felipe e Gabriel	Correção dos apontamentos.
19/11/19	4.6	Eunice G. Siqueira	Apontamentos na versão.
16/11/19	4.5	Carlos	Correção final da Fase 4.
15/11/19	4.4	Carlos, Felipe e Gabriel	Correção da Fase 4.
14/11/19	4.3	Carlos	Correções parciais da Fase 4.
07/11/19	4.2	Eunice G. Siqueira	Correção parcial da Fase 4.
02/11/19	4.1	Carlos, Felipe e Gabriel	Entrega da Fase 4
23/10/19	3.2	Carlos e Gabriel	Correção da Fase 3.
08/10/19	3.1	Eunice G. Siqueira	Correção da Fase 3.
07/09/19	3.0	Carlos, Felipe e Gabriel	Entrega da Fase 3.
12/08/19	2.4	Carlos, Felipe e Gabriel	Correção final da Fase 2.
30/06/19	2.3	Eunice G. Siqueira	Correção parcial da Fase 2.
22/06/19	2.2	Carlos, Felipe e Gabriel	Correção da Fase 2.
14/06/19	2.1	Eunice G. Siqueira	Correção parcial da Fase 2.
01/06/19	2.0	Carlos, Felipe e Gabriel	Entrega da Fase 2.
09/05/19	1.1	Eunice G. Siqueira	Correção parcial da Fase 1.
06/04/19	1.0	Carlos, Felipe e Gabriel	Entrega da Fase 1.

RESUMO

Neste documento apresentam-se as fases do Projeto Final de Curso *Eduq Student Life*, elaborado com a finalidade de desenvolver um sistema de software para escolas de Ensino Fundamental, seus professores, estudantes, pais ou responsáveis. Pesquisou-se sobre a Educação Básica, em especial, o Ensino Fundamental – Séries Iniciais – e aplicou-se a gerência de projeto tendo como referência o Guia de Conhecimentos do *Project Management Institute*. Realizou-se a engenharia de requisitos, a modelagem do sistema por meio da Linguagem de Modelagem Unificada e a programação com as tecnologias Java, Android e Google Firebase. Como resultado do projeto, obteve-se um sistema, nas plataformas *web* e *móvel*, que ao ser implantado pela instituição de ensino, contribuirá para a comunicação da escola com a família e o acompanhamento da vida escolar do aluno.

Palavras-chave: Educação Básica. Ensino Fundamental. Acompanhamento pedagógico. Sistemas de Informação.

LISTA DE FIGURAS

FIGURA 1 - Modalidades do aplicativo “Escola em Movimento”.....	21
FIGURA 2 - Telas de envio de mensagem aos responsáveis e monitoramento GPS.	22
FIGURA 3 - Grupos funcionais previstos e os níveis de decisão atingidos.....	25
FIGURA 4 - Partes interessadas alocadas em uma matriz de poder x interesse	29
FIGURA 5 - Ciclo de vida Incremental.....	30
FIGURA 6 - Quadro de atividades no Asana.	35
FIGURA 7 - Controle de versões BitBucket.	36
FIGURA 8 - Repositório da documentação.	37
FIGURA 9 - Níveis de visão de um banco de dados.....	62
FIGURA 10 - Trecho código gerador de sequência aleatórias de caracteres.....	67
FIGURA 11 - Método updateComponenteAlunoBoletins.	67
FIGURA 12 - Método busca dados conjunto de aviso enviado pela escola.	68
FIGURA 13 - Diagrama do sistema distribuído.....	71
FIGURA 14 - Paleta de cores utilizada no EduqWeb e EduqApp.	74
FIGURA 15 - Menu lateral do EduqWeb.....	75
FIGURA 16 - Lista de chamada no EduqWeb.	75
FIGURA 17 - Lista de classes abertas pela escola no EduQWeb.	76
FIGURA 18 - Mensagem de confirmação para a exclusão.	76
FIGURA 19 - Lista de classes cadastradas no EduqWeb.....	77
FIGURA 20 - Lista dos professores cadastrados no EduqWeb.	77
FIGURA 21 - Mensagem de alerta ao usuário.	78
FIGURA 22 - Técnica do breadcrumb aplicada pelo EduqWeb.	78
FIGURA 23 - Menu e atalhos.....	78
FIGURA 24 - Menu lateral do EduqWeb.....	79
FIGURA 25 - Página de autenticação do EduqWeb.	79
FIGURA 26 - Página para alterar a senha de acesso ao EduqWeb.	80
FIGURA 27 - Página de ajuda do EduqWeb.....	81
FIGURA 28 - Formulário para preenchimento de dúvidas no EduqWeb.	81
FIGURA 29 - Tela de autenticação do EduqApp.....	82
FIGURA 30 - Tela do perfil de usuário do EduqApp.	82
FIGURA 31 - Fluxos básicos e alternativos.....	86
FIGURA 32 - Tela de avisos e eventos do EduqApp.....	93
FIGURA 33 - Tela de acompanhamento EduqApp.....	93
FIGURA 34 - Tela de consulta do calendário escolar.....	94

LISTA DE QUADROS

QUADRO 1 - Níveis de ensinos no Brasil	18
QUADRO 2 - Esforços planejados e realizados por fase	34
QUADRO 3 - Estimativas de esforços de desenvolvimento	59
QUADRO 4 - Modelo físico do banco de dados	63
QUADRO 5 - Apresentação dos padrões utilizados	65
QUADRO 6 - Documento relevante para testes	83
QUADRO 7 - Equipamentos para a realização dos testes	84
QUADRO 8 - Softwares para a realização dos testes	84
QUADRO 9 - Identificação dos itens a serem testados	85
QUADRO 10 - Papéis e responsabilidades na implantação	89
QUADRO 11 - Treinamentos previstos	90
QUADRO 12 - Cronograma de atividades da implantação	90
QUADRO 13 - Documentos de apoio à implantação	91

LISTA DE ABREVIATURAS E SIGLAS

API	–	<i>Application Programming Interface.</i>
BNCC	–	Base Nacional Comum Curricular.
CMO	–	<i>Chief Marketing Officer.</i>
CEP	–	Código de Endereçamento Postal.
CPF	–	Cadastro Pessoa Física.
CSS	–	<i>Cascading Style Sheets.</i>
CRUD	–	<i>Create, Read, Update, Delete.</i>
DAO	–	<i>Data Access Object.</i>
DDL	–	<i>Data Definition Language.</i>
DML	–	<i>Data Manipulation Language.</i>
EAP	–	Estrutura Analítica de Projeto.
FAI	–	FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação.
FCM	–	<i>Firebase Cloud Messaging</i>
GPS	–	<i>Global Positioning System.</i>
HCI	–	<i>Human Computer Interaction.</i>
HTML	–	<i>Hypertext Markup Language.</i>
HTTPS	–	<i>Hyper Text Transfer Protocol Secure.</i>
JAR	–	<i>Java Archive.</i>
JEE	–	<i>Java Enterprise Edition</i>
MEC	–	Ministério da Educação.
MVC	–	<i>Model, View, Controller.</i>
NoSQL	–	<i>Not Only Structured Query Language</i>
PCU	–	Pontos de Casos de Uso.
PF	–	Pontos de Função.
PMI	–	<i>Project Management Institute.</i>
SBC	–	Sociedade Brasileira de Computação.
SGBD	–	Sistema de Gerenciamento de Banco de Dados.
SSL	–	<i>Secure Sockets Layer.</i>
PFC	–	Projeto Final de Curso.
TIC	–	Tecnologia de Informação e Comunicação.

TLS	–	<i>Transport Layer Security.</i>
UF	–	Unidade da Federação.
UML	–	<i>Unified Modeling Language.</i>
WBS	–	<i>Work Breakdown Structure.</i>
VPS	–	<i>Virtual Private Server.</i>

SUMÁRIO

1 INTRODUÇÃO	14
2 REVISÃO BIBLIOGRÁFICA	17
2.1 EDUCAÇÃO BÁSICA.....	17
2.1.1 Relato de uma profissional do Ensino Fundamental I	18
2.2 TECNOLOGIAS DE INFORMAÇÃO E COMUNICAÇÃO APLICADAS À ROTINA ESCOLAR	19
2.2.1 O uso dos recursos tecnológicos na sala de aula	20
2.3 TRABALHOS RELACIONADOS	20
3 OBJETIVO DO PROJETO.....	23
3.1 FORMULAÇÃO DO PROBLEMA	23
3.2 OBJETIVOS	24
3.3 JUSTIFICATIVA	24
3.4 PÚBLICO ALVO	24
3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS.....	25
3.6 MODELO DE NEGÓCIO PROPOSTO.....	26
3.7 CONSIDERAÇÕES SOBRE OS OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL (ODS).....	26
4 GERÊNCIA DO PROJETO	28
4.1 PLANO DE PROJETO.....	28
4.1.1 Partes Interessadas	28
4.1.2 Modelo de Ciclo de Vida	30
4.1.3 Recursos Necessários.....	30
4.2 ÁREAS DE CONHECIMENTO	32
4.2.1 Gestão do Escopo.....	32
4.2.2 Gestão do Tempo	33
4.2.3 Gestão da Integração.....	35
4.2.4 Gestão da Qualidade	37
4.2.5 Gestão dos Riscos.....	38
5 ESPECIFICAÇÃO E ANÁLISE DE REQUISITOS	39
5.1 DESCRIÇÃO DE REQUISITOS	39
5.1.1 Requisitos Funcionais.....	39
5.1.2 Requisitos Não Funcionais.....	55
5.2 ANÁLISE DOS REQUISITOS	58
5.2.1 Modelo de Casos de Uso.....	58
5.2.2 Modelo Conceitual dos Dados	58

5.2.3 Modelo Inicial da Interface de Usuário	58
5.3 MÉTRICAS PARA ESTIMATIVA DE ESFORÇO	59
6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE	60
6.1 VISÃO ESTRUTURAL	60
6.1.1 Diagrama de Pacotes	60
6.1.2 Diagramas de Classes	60
6.1.3 Diagramas de Objetos	61
6.2 VISÃO COMPORTAMENTAL	61
6.2.1 Projeto das Interações entre Objetos	61
6.2.2 Diagrama de Atividades	62
6.3 VISÃO DOS DADOS	62
6.3.1 Modelo Lógico	63
6.3.2 Dicionário de Dados do Modelo Lógico	63
6.3.3 Estrutura Física do Banco de Dados	63
6.4 VISÃO FÍSICA	63
6.4.1 Diagrama de Componentes	63
6.4.2 <i>Frameworks</i> Adotados	64
6.4.3 <i>Design Patterns</i> Aplicados	64
6.4.4 Convenções e Guias para Codificação	65
6.4.5 Análise de Complexidade Algorítmica	66
6.5 PROJETO DE SISTEMA DISTRIBUÍDO	68
6.5.1 Procedimentos para Tratamento dos Desafios	69
6.5.2 Tecnologias e Arquiteturas de Distribuição	71
6.5.2.1 Google Firebase	72
6.6 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR	72
6.6.1 Perfil de Usuário	73
6.6.2 Aspecto Visual da Interface de Usuário	73
6.6.3 Heurísticas de Usabilidade	75
7 PLANO DE TESTES	83
7.1 FINALIDADE	83
7.2 ESCOPO	83
7.2.1 Referências a Documentos Relevantes	83
7.2.2 Ambiente para a Realização dos Testes	84
7.3 ESPECIFICAÇÃO DOS CASOS DE TESTE	84
7.3.1 Item a Testar	85
7.3.2 Rastreabilidade entre Requisitos e Casos de Teste	85

7.3.3 Descrição dos Casos de Teste.....	85
7.4 RESULTADOS DOS TESTES	86
7.4.1 Histórico de Realização	86
7.4.2 Resultados Obtidos	87
8 PLANO DA IMPLANTAÇÃO.....	88
8.1 METODOLOGIA	88
8.1.1 Descrição da Metodologia	88
8.1.2 Matriz de Responsabilidade	89
8.2 TREINAMENTO PREVISTO.....	89
8.3 CRONOGRAMA DE IMPLANTAÇÃO	90
8.4 DOCUMENTOS DE APOIO À IMPLANTAÇÃO	91
8.5 VISÃO DA IMPLANTAÇÃO	91
9 CONCLUSÃO.....	92
REFERÊNCIAS	95
OBRAS CONSULTADAS	97
APÊNDICE A – EAP	99
APÊNDICE B – DIAGRAMAS DE CASOS DE USO	100
APÊNDICE C – DIAGRAMA ENTIDADE RELACIONAMENTO	101
APÊNDICE D – PLANILHA DOS RISCOS NEGATIVOS	102
APÊNDICE E – RELATÓRIO DE DESEMPENHO	103
APÊNDICE F - MODELO INICIAL DA INTERFACE DE USUÁRIO	104
APÊNDICE G - DIAGRAMA DE PACOTES	105
APÊNDICE H - DIAGRAMA DE CLASSES.....	106
APÊNDICE I- DIAGRAMA DE OBJETOS	107
APÊNDICE J - DIAGRAMAS DE SEQUÊNCIA	108
APÊNDICE K – DESCRIÇÕES DOS CASOS DE USO.....	109
APÊNDICE L - DIAGRAMA DE VISÃO GERAL DA INTERAÇÃO.....	110
APÊNDICE M - DIAGRAMA DE ATIVIDADE.....	111
APÊNDICE N - MODELO LÓGICO	112
APÊNDICE O - DICIONÁRIO DE DADOS DO MODELO LÓGICO	113
APÊNDICE P – PERFIL DO USUÁRIO.....	114
APÊNDICE Q – LISTA DE VERIFICAÇÃO DA DOCUMENTAÇÃO.....	115
APÊNDICE R – CRONOGRAMA E DICIONARIO EAP.....	116
APÊNDICE S – ESTIMATIVA DE ESFORÇO	117

APÊNDICE T – REQUISIÇÕES DE MUDANÇA	118
APÊNDICE U – DIAGRAMA DE COMPONENTES.....	119
APÊNDICE V – ESTRUTURA FISICA DO BANCO DE DADOS	120
APÊNDICE W – RASTREABILIDADE ENTRE REQUISITOS E CASOS DE TESTE	121
APÊNDICE X - DESCRIÇÃO DOS CASOS DE TESTE.....	122
APÊNDICE Y - HISTÓRICO DE REALIZAÇÃO	123
APÊNDICE Z – MANUAL DE USUÁRIO.....	124
APÊNDICE AA – DIAGRAMA DE IMPLANTAÇÃO DA UML.	125

1 INTRODUÇÃO

Para que o Brasil seja protagonista no cenário mundial e sua população atinja os melhores patamares de qualidade de vida, diminuindo as desigualdades sociais e econômicas, o caminho é por meio da educação, do desenvolvimento da capacidade criativa e da inovação (BRASIL, 2018).

No Brasil, a Educação Básica é o primeiro nível de ensino escolar formal, sendo obrigatória para crianças e adolescentes entre 4 e 17 anos. Esse nível contém 3 etapas: Educação Infantil (para crianças com menos de 6 anos), Ensino Fundamental (para crianças e adolescentes entre 6 e 14 anos) e Ensino Médio (para adolescentes dos 15 aos 17 anos). O Ensino Fundamental, por envolver especificidades e demandas pedagógicas diferentes, subdivide-se em Séries Iniciais - do 1º. aos 5º. anos - e Anos Finais - do 6º. aos 9º. anos. Essa subdivisão é também conhecida por Ensino Fundamental I e Ensino Fundamental II, respectivamente.

O Ensino Fundamental I, do 1º aos 5º anos, envolve crianças a partir dos 6 anos de idade. Nessa fase de escolarização, o aluno começa seu processo de alfabetização, quando se formam as bases do seu conhecimento para toda vida. É uma etapa de transição da Educação Infantil e a criança precisa de cuidado e de respeito para uma construção mental, motora e emocional saudável. Também requer ser acompanhada de seus pais/responsável de perto e de auxílio em sua vida escolar.

Conforme Brasil (2018), todo esse quadro impõe desafios às instituições de ensino para o cumprimento do seu papel em relação à formação educacional das novas gerações. Nesse contexto, é importante que a escola aproveite o potencial da comunicação no universo digital, pelo qual ela poderá instituir novos modos de promover a aprendizagem, a interação e o compartilhamento de significados entre professores, alunos e famílias.

Na maior parte das instituições de ensino, os professores e supervisores se comunicam com os responsáveis pelas crianças por meio de um caderno/agenda de recados, próprio de cada criança, no qual a escola anota e recebe informações. Porém, esse método pode ser falho, pois é comum o extravio do caderno ou o esquecimento dele pela criança ou responsável.

O presente projeto, nomeado *EduQ Student Life*, objetiva o desenvolvimento de um sistema de *software* composto por componentes *web*, *mobile* (móvel) e de serviços, que proporcione aos professores e supervisores da instituição de ensino (no caso, Ensino Fundamental I) a capacidade de registrar os avisos e atividades escolares realizadas ou previstas. Os responsáveis pelas crianças poderão acessar esses avisos e se comunicarem com a escola por meio do sistema. Adicionalmente, o sistema proporcionará consultas e relatórios aos usuários, permitindo um melhor acompanhamento tanto por parte do responsável pela criança, quanto pelos professores e colaboradores da escola.

Neste documento, além deste capítulo de introdução, o leitor encontra os capítulos que são apresentados a seguir.

No Capítulo 2, o leitor encontra a revisão bibliográfica, elaborada a partir de um estudo de obras, teses e dissertações, para explorar o assunto em questão, uma pesquisa feita com uma profissional da área e trabalhos relacionados ao presente projeto.

No Capítulo 3, o leitor encontra os objetivos do projeto, os problemas operacionais enfrentados pelas instituições de ensino, a justificativa deste projeto, o público alvo a quem se destina e os níveis de decisão e grupos funcionais atendidos.

No Capítulo 4, o leitor encontra todos os processos necessários para realizar o planejamento e gerenciamento deste projeto, como o plano de projeto, as partes interessadas, o modelo de ciclo de vida e os recursos necessários para o desenvolvimento. Encontra também as áreas do conhecimento e suas ferramentas e processos, como a estrutura analítica do projeto (EAP), o dicionário EAP, a lista de atividades, diagrama de rede, cronograma de atividades, o controle das mudanças, riscos e outros artefatos.

No Capítulo 5, o leitor encontra a especificação e análise de requisitos funcionais e não funcionais, a visão funcional, modelada por meio do Modelo de Casos de Uso, a visão de dados, modelada por meio do Modelo Entidade-Relacionamento, e as estimativas de esforços.

No Capítulo 6, o leitor encontra a arquitetura e projeto do sistema. A arquitetura está representada pela visão estrutural, por meio do diagrama de pacotes, diagrama de classes e diagrama de objetos; pela visão comportamental, por meio do diagrama de sequência,

diagrama de visão geral de interação, diagrama de atividades; e pela visão de dados, por meio do modelo lógico e dicionário de dados. Constam também os padrões de projeto, projeto de sistemas distribuídos e também o projeto da interação humano computador que são representados pelos aspectos visuais e as heurísticas de usabilidade.

No Capítulo 7, o leitor encontra o plano de teste elaborado para o EduQ *Student Life*. Neste capítulo são especificados o tipo de teste realizado, os documentos relevantes para a elaboração do plano, o ambiente para a realização dos testes e os itens que serão testados. Além disso, o capítulo também é composto pelo relato dos resultados dos testes realizados.

No Capítulo 8, encontra-se o planejamento da implantação do sistema de software em seu ambiente de operacional.

Por fim, no Capítulo 9, encontra-se a conclusão do projeto.

Este documento contém, ainda, as referências, obras consultadas e apêndices.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo encontram-se os embasamentos referentes à Educação Básica, o relato de uma profissional do Ensino Fundamental I e trabalhos relacionados ao EduQ *Student Life*.

2.1 EDUCAÇÃO BÁSICA

A educação é um direito fundamental compartilhado entre Estado, família e sociedade. Por meio dela, acontece o pleno desenvolvimento da pessoa, seu preparo para o exercício da cidadania e sua qualificação para o trabalho (BRASIL, 1998).

A Educação Básica é o primeiro nível de ensino escolar formal, sendo obrigatória para crianças e adolescentes entre 4 e 17 anos. De acordo com a Base Nacional Comum Curricular (BNCC), publicada pelo Ministério da Educação (MEC), em 2018, a Educação Básica visa à formação e o desenvolvimento humano global, o que implica em compreender a complexidade e a não linearidade desse desenvolvimento. Significa, também, assumir uma visão integral do sujeito de aprendizagem e entender a escola como espaço privilegiado de aprendizagem e inclusão.

A Educação Básica contém 3 etapas: Educação Infantil (para crianças com menos de 6 anos), Ensino Fundamental (para crianças e adolescentes entre 6 e 14 anos) e Ensino Médio (para adolescentes dos 15 aos 17 anos). O Ensino Fundamental, por envolver especificidades e demandas pedagógicas diferentes e englobar competências e habilidades a serem desenvolvidas durante essa etapa complexa, subdivide-se em Séries Iniciais - do 1º. aos 5º. anos – e Anos Finais – do 6º. aos 9º. anos. Essa subdivisão é também conhecida por Ensino Fundamental I e II, respectivamente mostrada no Quadro 1.

Nível de ensino	Ano escolar	Idade regular
Educação Infantil	--	Menos de 6 anos
Ensino Fundamental I	1º ano	6 anos
	2º ano	7 anos
	3º ano	8 anos
	4º ano	9 anos

Continua

	5º ano	10 anos
Ensino Fundamental II	6º ano	11 anos
	7º ano	12 anos
	8º ano	13 anos
	9º ano	14 anos

QUADRO 1 - Níveis de ensinos no Brasil

Fonte: Brasil (2018)

É no Ensino Fundamental I que o aluno passa a ter uma noção maior da complexidade de vários assuntos e inicia a visão sobre a realidade com diferentes perspectivas. A viabilidade de se colocar no lugar do outro se expande, a procura por especificações sociais e pela justiça torna-se, em muitos casos, uma obrigação. Por tudo isso, os estudos ganham maior profundidade e grau de formalização, com volta constante aos mesmos conteúdos, a fim de analisá-los, registrá-los, aprofundá-los, exercitá-los e generalizá-los para outros contextos (MIRO, 2019).

O aluno nessa fase deve compreender, construir, divulgar e vivenciar valores na rotina escolar, favorecendo a realização da cidadania, na busca de qualidade de vida, promover o desenvolvimento integral através da construção de conhecimentos e habilidades, aprendendo a aprender, considerar as diferentes opiniões dos grupos, buscando contínua atualização e qualificação (MIRO, 2019).

2.1.1 Relato de uma profissional do Ensino Fundamental I

Jéssica Yara Rodrigues é uma profissional que atua em uma escola com Ensino Fundamental I. Ela trabalha na Escola Municipal Coronel Joaquim Inácio, no município de Santa Rita do Sapucaí – MG. Em uma entrevista concedida em março de 2019, ela relatou como são preparadas as atividades com os alunos, como os responsáveis pelas crianças são avisados sobre as entregas dos boletins, reuniões, tarefas para casa, entre outros. A seguir o relato apresentado pela professora:

As atividades diárias são planejadas individualmente pelo professor fora do horário de aula, porém, o que é preciso trabalhar durante o bimestre ou durante a semana, é definido durante o módulo dos professores. Essas não são informadas

aos pais e o registro é feito nos cadernos de cada disciplina ou tarefa. No que diz respeito às atividades avaliativas, são duas, as avaliações e os trabalhos. As avaliações estão sendo elaboradas nos módulos por escolaridade, quando todos os professores das escolas municipais se reúnem por ano/nível que lecionam, então todas turmas de determinado ano fazem a mesma avaliação. Os trabalhos avaliativos são elaborados individualmente pelo professor, sendo que ele pode escolher como será desenvolvido com os alunos, em sala de aula ou em casa. Especificamente, as avaliações são informadas aos pais por meio de bilhetes, que vão no caderno de recado ou de tarefa. Esses bilhetes devem ser assinados por um dos responsáveis pela criança. Os registros de notas vão no diário escolar, que fica na instituição, e no boletim dos alunos, que são entregues a cada fim de bimestre nas reuniões com os responsáveis (RODRIGUES, 2019, p.1).

Percebe-se, pelo relato da profissional, que a maior parte das rotinas de comunicação escola-família é manual e demanda tempo operacional dos colaboradores da escola.

A professora Jéssica foi questionada se um sistema automatizado de comunicação da escola com os responsáveis pelas crianças ajudaria e agregaria à escola e aos colaboradores:

Ajudaria muito, principalmente, na questão de comunicação com os pais/responsáveis, pois muitas vezes temos alunos que se esquecem de mostrar o bilhete aos pais/responsáveis ou não o mostram, propositalmente, dificultando o processo educacional. Um sistema de apoio à comunicação entre a escola e os pais/responsáveis facilitaria muito na questão de comunicação e na resolução de problemas ou dúvidas que podem vir aparecer (RODRIGUES, 2019, p.1).

2.2 TECNOLOGIAS DE INFORMAÇÃO E COMUNICAÇÃO APLICADAS À ROTINA ESCOLAR

Para as escolas e educadores, a necessidade criada pelo uso da Tecnologia da Informação e Comunicação (TICs), é saber como aplicar todo o potencial existente no sistema educacional, especialmente nos seus componentes pedagógicos e processos de ensino e de aprendizagem.

Libâneo (2007, p. 309) afirma que: “o grande objetivo das escolas é a aprendizagem dos alunos, e a organização escolar necessária é a que leva a melhorar a qualidade dessa aprendizagem”.

Para Moran (2000, p. 63), o ensino com as novas mídias “será uma revolução se mudarmos simultaneamente os paradigmas convencionais do ensino, que mantêm distantes professores e alunos. Caso contrário, conseguiremos dar um verniz de modernidade, sem mexer no essencial”.

2.2.1 O uso dos recursos tecnológicos na sala de aula

Demo (2017, p. 1), sobre as Tecnologias de Informação e Comunicação, aponta que “proposta que investe na introdução das TICs na escola só pode dar certo passando pelas mãos dos professores. O que transforma tecnologia em aprendizagem, não é a máquina, o programa eletrônico, o *software*, mas o professor, em especial em sua condição socrática”.

As tecnologias estão mais presentes em todos os ambientes. Na maior parte das escolas, professores e alunos utilizam TV, vídeo, DVD, rádio, computadores e a *Internet* na prática pedagógica, ajudando a tornar o processo ensino aprendizagem mais significativo.

As mídias têm grande poder pedagógico visto que se utilizam da imagem. Assim, torna-se cada vez mais necessário que a escola se aproprie dos recursos tecnológicos, dinamizando o processo de aprendizagem.

Devemos considerar como ideal um ensino usando diversos meios, um ensino no qual todos os meios deveriam ter oportunidade, desde os mais modestos até os mais elaborados: desde o quadro, os mapas e as transparências de retroprojeto até as antenas de satélite de televisão. Ali deveriam ter oportunidade também todas as linguagens: desde a palavra falada e escrita até as imagens e sons, passando pelas linguagens matemáticas, gestuais e simbólicas (SANCHO, 2001, p. 136).

Assim, a tecnologia educacional deve estar presente nas escolas para melhoria do processo ensino aprendizagem.

2.3 TRABALHOS RELACIONADOS

Como trabalho relacionado ao presente projeto, cita-se um sistema *Web* de gestão para Educação Infantil, chamado *Emother*, elaborado por alunos do curso de Sistemas de Informação, em 2015. O sistema permite aos professores e demais profissionais da Educação, o registro das atividades diárias realizadas pelos alunos, objetivando melhor administração

da informação sobre cada criança, libertando, se desejado for, dos registros manuais sobre as tarefas diárias em cadernos ou agendas (DOMICIANO JÚNIOR; OLIVEIRA; CASTRO, 2015).

Outro sistema relacionado encontrado foi o “Escola em movimento”, que é um aplicativo escolar que oferece um meio de comunicação entre as instituições de ensino, os pais, alunos e colaboradores, via *smartphones*, *tablets* ou computadores. A empresa desenvolvedora desse aplicativo foi fundada em Belo Horizonte (MG), em 2013, pelos cofundadores Guilherme Rocha e Leonardo Silva (ESCOLA EM MOVIMENTO, 2016).

Segundo o site do “Escola em Movimento” são atendidas escolas com 10 até 15 mil crianças. A empresa divulga o modelo de licenciamento somente por meio da sua equipe de consultores. Algumas telas do aplicativo são mostradas nas Figuras 1 e 2. Além das funcionalidades de envio de mensagens, calendários e agendas, o aplicativo também apresenta uma “carteirinha estudantil” e diários da turma.



FIGURA 1 - Modalidades do aplicativo “Escola em Movimento”.

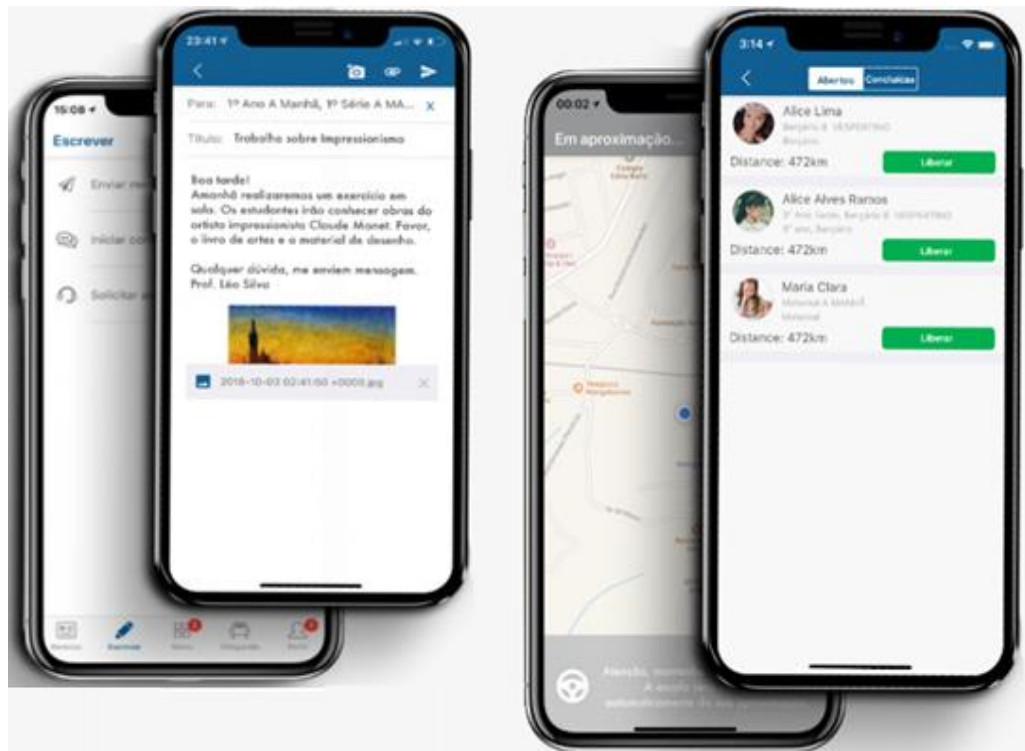


FIGURA 2 - Telas de envio de mensagem aos responsáveis e monitoramento GPS.

Fonte: Escola em Movimento (2019)

3 OBJETIVO DO PROJETO

Este capítulo relata a formulação do problema, pela qual se mostra a necessidade do projeto *EduQ Student Life*, os objetivos gerais e específicos, a justificativa para seu desenvolvimento, o público alvo a quem se destina, os grupos funcionais e níveis de decisão envolvidos.

3.1 FORMULAÇÃO DO PROBLEMA

O envio de recados em papel é adotado pela maior parte das escolas de Ensino Fundamental I para informar aos pais (ou outro responsável pela criança), quando há uma atividade a ser feita fora do horário de aula. O aluno deve portar um caderno de recados no qual o professor escreve avisos ou fixa bilhetes para o responsável, a fim de avisá-lo sobre as atividades escolares, como reuniões, provas e passeios. Muitas vezes, a criança perde esse tipo de registro ou o joga fora, o que faz com que a comunicação entre família e escola fique prejudicada.

Diante da importância da Educação Básica e, especialmente, do Ensino Fundamental I, os meios de comunicação entre família-escola deve ser o mais eficiente possível. A relação entre família e escola deve ser fundamentalmente dialógica, uma vez que o diálogo e a colaboração entre todos os agentes do contexto educativo são ferramentas essenciais para promover um ambiente equilibrado, pacífico e harmonioso de aprendizagem.

Em relação ao uso das TICs, a BNCC explica que as experiências das crianças em seu contexto familiar, social e cultural, suas memórias, seu pertencimento a um grupo e sua interação com as mais diversas TICs são fontes que estimulam sua curiosidade e a formulação de perguntas. Além disso, em decorrência do avanço e da multiplicação das tecnologias de informação e comunicação e do crescente acesso a elas pela disponibilidade de computadores, telefones celulares, *tablets* e afins, os estudantes estão dinamicamente inseridos na cultura digital (BRASIL, 2018).

Diante desse contexto, observa-se que as tecnologias digitais podem contribuir muito para comunicação da escola com os pais ou responsável, abrindo caminhos para que aplicativos *web* e *mobile* (móvel) possam ser desenvolvidos para melhorar a produtividade

dos colaboradores, a relação família-escola e, por conseguinte, o desempenho estudantil da criança.

3.2 OBJETIVOS

O propósito geral deste projeto é o desenvolvimento de um sistema de *software*, com componentes *web*, *mobile* e *web services*, para apoiar a comunicação entre a escola e os responsáveis pelos alunos matriculados em classes do Ensino Fundamental I.

Os objetivos específicos são:

- a) estudar sobre o sistema educacional brasileiro, em particular, o Ensino Fundamental – anos iniciais;
- b) desenvolver um aplicativo *web* por meio do qual o profissional da Educação conseguirá gerar boletins escolares e divulgar comunicados aos responsáveis pelo aluno;
- c) desenvolver um aplicativo *mobile* por meio do qual o responsável pelo aluno conseguirá acompanhar as atividades escolares em curso.

3.3 JUSTIFICATIVA

O EduQ *Student Life* pretende melhorar a forma pela qual a escola irá se comunicar com os pais ou responsável pelos alunos do Ensino Fundamental I, considerando que, muitas vezes, esse processo é feito de forma manual, em papel e, por vezes, sem eficiência.

O sistema deverá proporcionar uma melhor forma de comunicação de datas de provas e reuniões, entregas de boletim e demais atividades fora da sala de aula. Espera-se que o sistema melhore a participação dos pais nas tarefas para casa, aumentando o nível de comprometimento da criança dentro da sala de aula, crescendo a inspiração do aluno ao ter os pais/responsável mais próximos e elevando o nível de aprendizagem escolar.

3.4 PÚBLICO ALVO

O sistema de *software* foca em auxiliar várias instituições com Ensino Fundamental I em Santa Rita do Sapucaí (MG), nas quais foram coletadas informações para o

desenvolvimento deste projeto. Sendo assim, o público alvo são as escolas cujos métodos utilizados para comunicação com os pais/responsável sejam manuais e que necessitam de uma ferramenta computacional para auxiliar na comunicação com a família. Da mesma forma, são públicos também os pais/responsável de alunos do Ensino Fundamental I que tenham acesso a dispositivos móveis com acesso à Internet de banda larga.

3.5 NÍVEIS DE DECISÃO E GRUPOS FUNCIONAIS ATENDIDOS

De acordo com Bateman e Snell (1998), dentro das organizações existem três níveis hierárquicos: estratégico, tático e operacional, de acordo com o tipo de trabalho que é desenvolvido em cada nível.

O nível estratégico compreende os altos executivos da organização responsáveis pela definição dos objetivos e tomada de decisões quanto às questões de longo prazo, como para sobrevivência, crescimento e eficácia geral. O planejamento, em nível tático, é utilizado para traduzir os objetivos gerais e as estratégias da alta diretoria em objetivos e atividades mais específicos. O principal desafio nesse nível é promover um contato eficiente e eficaz entre o nível estratégico e o nível operacional. Já no nível operacional, o processo é de uma menor amplitude, com foco em trabalhar junto aos colaboradores, implementando os planos específicos definidos no planejamento tático (BATEMAN; SNELL,1998)

Atingindo o escopo do projeto, os níveis de decisão e os grupos funcionais são mostrados na Figura 3.

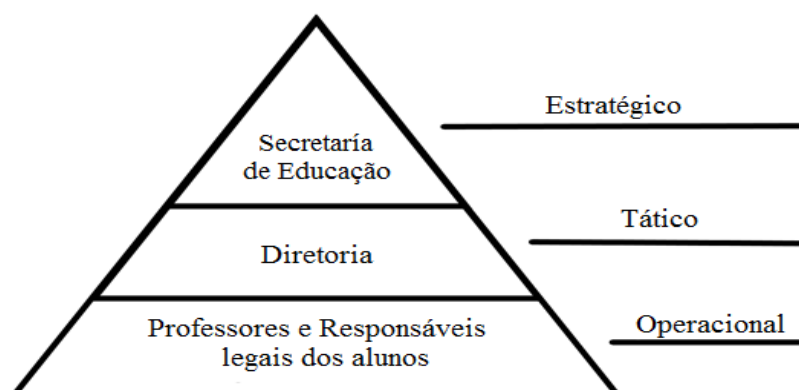


FIGURA 3 - Grupos funcionais previstos e os níveis de decisão atingidos.

Neste projeto, o sistema de *software* a ser desenvolvido deverá atuar apenas em nível operacional.

3.6 MODELO DE NEGÓCIO PROPOSTO

O modelo de negócio proposto para o *EduQ Student Life* consiste em oferecer o sistema como *Software as a Service* (SaaS) que tem como premissa uma forma de disponibilizar *softwares* e soluções de tecnologia por meio da Internet, como um serviço. Com esse modelo, não é necessário instalar, manter e atualizar *hardwares* no ambiente local. Quanto ao *software*, apenas se faz necessário para um usuário interessado, a instalação de um aplicativo móvel. Para isso, cada instituição de ensino interessada terá acesso a uma conta administrativa, na qual poderá gerenciar aspectos referentes aos alunos matriculados, como, seus boletins e comunicados. Um responsável terá acesso, via aplicativo, a uma conta na qual terá a possibilidade de acompanhar a vida escolar de um aluno sob sua responsabilidade, mediante consultas aos dados apontados pela escola.

3.7 CONSIDERAÇÕES SOBRE OS OBJETIVOS DO DESENVOLVIMENTO SUSTENTÁVEL (ODS)

Em setembro de 2015, mais de 150 líderes mundiais estiveram reunidos em Nova York a fim de discutirem uma agenda de desenvolvimento sustentável para os próximos anos. Ao findar desta reunião, houve um consenso dos líderes que ali estiveram sobre 17 objetivos de desenvolvimento sustentável que deveriam ser trabalhados por todos os países até o ano de 2030.

Esses objetivos se constituem em metas para a sustentabilidade do planeta e, conseqüentemente, da humanidade. Destacam-se temas como: pobreza, fome, igualdade de gênero e educação.

O *EduQ Student Life* adota um olhar sensível quanto às necessidades de mudança do planeta e, com isso, compromete-se com o auxílio para alcançar:

- a) Objetivo 4 - Educação de qualidade: assim como o propósito deste projeto, uma educação de qualidade passa pelo acompanhamento dos alunos por seus pais ou responsáveis, de forma que, quanto mais próximo um responsável

estiver da vida escolar de um aluno, melhor pode ser seu rendimento e absorção de conhecimentos;

- b) **Objetivo 11 – Cidades e comunidades sustentáveis:** para que uma comunidade seja sustentável, faz-se necessário romper o obstáculo da exclusão. Neste passo, o *EduQ Student Life* vem para auxiliar no âmbito de fornecer um meio no qual possa haver a inclusão na esfera educacional, onde os responsáveis estarão mais próximos das escolas e professores, bem como mais influentes na vida escolar dos alunos.

4 GERÊNCIA DO PROJETO

Neste capítulo estão descritos e ilustrados todos os processos necessários para realizar o planejamento e gerenciamento do projeto.

Gerência do Projeto pode ser definida como “a aplicação de conhecimentos, habilidades, e técnicas para projetar atividades que visem atingir ou exceder as necessidades e expectativas das partes envolvidas, com relação ao projeto” (PROJECT MANAGEMENT INSTITUTE, 2008, p. 6).

4.1 PLANO DE PROJETO

O planejamento do projeto está descrito levando-se em consideração as partes interessadas e os recursos humanos, de *hardware* e *software* necessários para desenvolvimento do projeto.

4.1.1 Partes Interessadas

O projeto possui cinco partes interessadas, cada uma com seus interesses específicos para realização deste projeto. São elas:

- a) Equipe de desenvolvedores: composta pelos integrantes Carlos Christian Zanin, Felipe Leme Simões Balestra e Gabriel Rodrigo Martins da Silva. Sendo o objetivo, construir o sistema de *software* *EduQ Student Life* de acordo com os requisitos para eles definidos.

Interesse: alto. Poder: alto.

- b) Orientadores: compostos pelo professor Afonso Celso Soares e a professora Eunice Gomes de Siqueira com o papel de orientar este projeto de conclusão do curso Sistemas de Informação.

Interesse: alto. Poder: baixo.

- c) Instituição de Ensino Fundamental I: a instituição de ensino Escola Municipal Coronel Joaquim Inácio dará apoio a elaboração do EduQ *Student Life*, contribuindo com informações pertinentes aos requisitos a serem atendidos.

Interesse: baixo. Poder: baixo.

- d) Profissional da Educação: os professores responsáveis pelos alunos em sala de aula.
- e) Responsável: pais (ou outro responsável legal) da criança matriculada no Ensino Fundamental I, os quais poderão acompanhar todas as atividades escolares, saber sobre as reuniões, eventos da escola, e eventuais acontecimentos no dia a dia da criança pela qual é responsável.

Interesse: alto. Poder: baixo

A Figura 4 demonstra de forma gráfica o nível de interesse e a responsabilidade de cada parte interessada no EduQ *Student Life*.

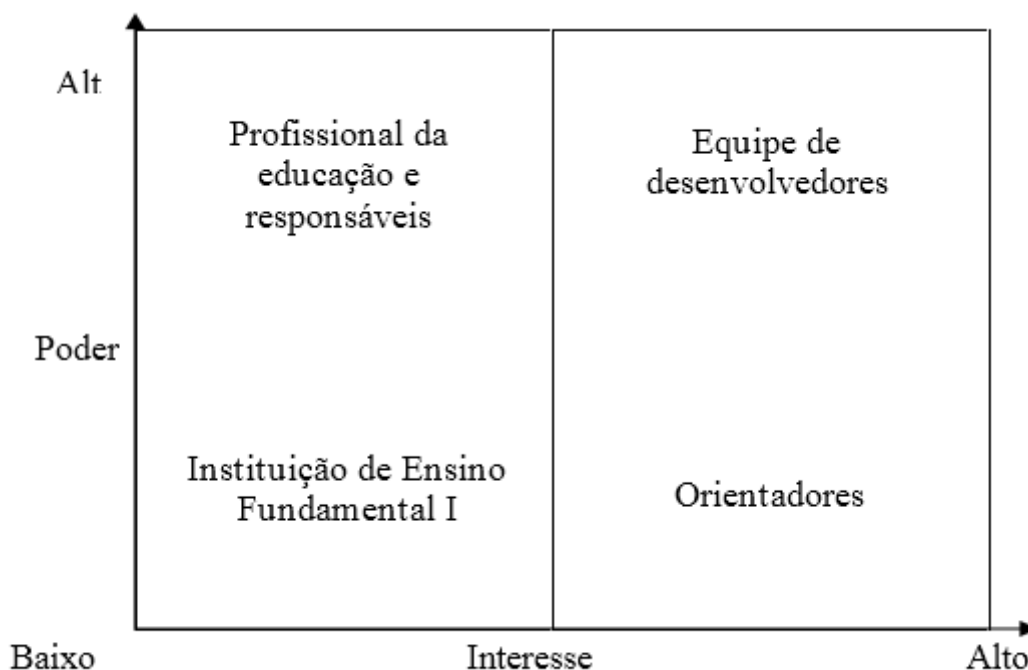


FIGURA 4 - Partes interessadas alocadas em uma matriz de poder x interesse

4.1.2 Modelo de Ciclo de Vida

O modelo de ciclo de vida que norteia o desenvolvimento do *Eduq Student Life* é o modelo Incremental, sendo que a cada entrega é retomada a fase de planejamento da próxima entrega passando pelas fases posteriores. O projeto está dividido em quatro entregas, sendo a primeira em 06 de abril de 2019, a segunda, em 01 de junho de 2019, a terceira, em 07 de setembro de 2019, e a última entrega prevista para o dia 02 de novembro de 2019. Na imagem a seguir (Figura 5) é mostrado o ciclo de vida Incremental.

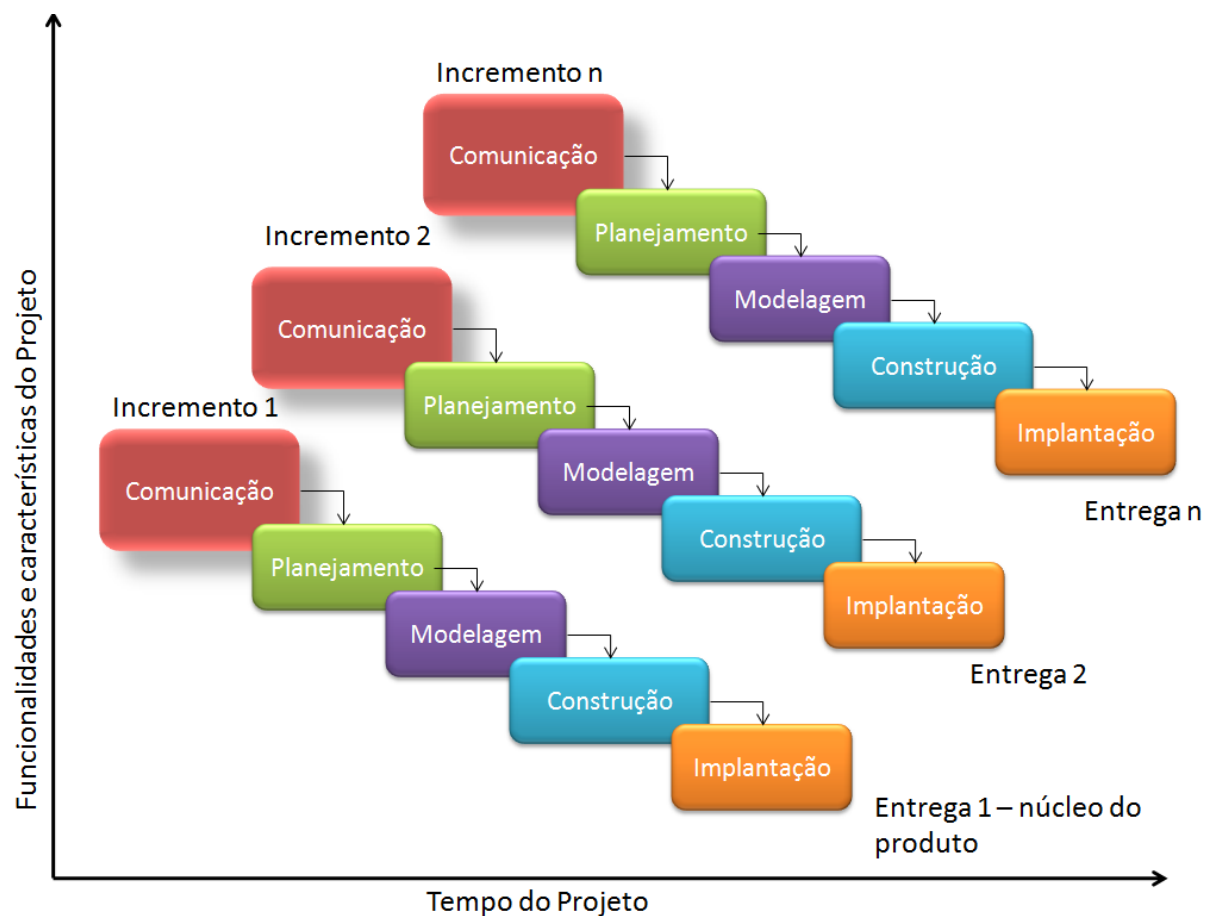


FIGURA 5 - Ciclo de vida Incremental

Fonte: Pressman (2006).

4.1.3 Recursos Necessários

A equipe é formada pelos integrantes Carlos Christian Zanin, graduando do curso de Sistemas de Informação, natural de Lorena-SP; Felipe Leme Simões Balestra, graduando do curso de Sistemas de Informação, natural de Santa Rita do Sapucaí-MG e Gabriel Rodrigo

Martins da Silva, graduando do curso de Sistemas de Informação, natural de Moji Mirim - SP.

Para que seja desenvolvido o projeto, a equipe possui como orientadores a professora Eunice Gomes de Siqueira e o professor Afonso Celso Soares. Este projeto tem natureza interdisciplinar e para orientações pertinentes à área de Educação Básica conta com o apoio da professora Jéssica Yara Rodrigues Silva.

Para desenvolvimento do projeto, devem ser usados os seguintes recursos de software:

- a) Microsoft Word 2016;
- b) Microsoft PowerPoint 2016;
- c) Sistema Operacional Microsoft Windows 10 Professional 64 bits;
- d) Netbeans 8.2;
- e) PostgreSQL 9.3;
- h) WBS Tools versão Web;
- i) Microsoft Project 2016;
- j) Microsoft Visio 2016;
- k) Oracle VirtualBox 5.2.6;
- l) Sistema Operacional MacOS Mojave;
- m) Sistema Operacional MacOS HighSierra;
- n) Android Studio 3.3.1;
- o) Draw.io versão 10.6.5;
- p) Firebase;

- q) Asana Android versão 7;
- r) Asana versão Web;
- s) Visual Paradigm *Community Edition*.

Os recursos de hardware são:

- a) MacBook Pro Mid 2009 - processador Intel Core 2 Duo, 2.53 GHz, 64-bit (x64); 4 Gbytes RAM e 250 Gbytes de armazenamento;
- b) MacBook Air 2017 - processador Intel Core i5, 1.8 GHz, 64-bit (x64); 8 Gbytes RAM e 128 Gbytes de armazenamento;
- c) Notebook Lenovo - processador Intel(R) Core (TM) i5- 3360M, CPU 2.80GHz, 8 Gbytes RAM e 400 Gbytes de armazenamento;
- d) *Smartphone* - Motorola Moto G4, com 2 GBytes de RAM e 32 GBytes de armazenamento.

4.2 ÁREAS DE CONHECIMENTO

Nesta seção detalham-se as áreas de conhecimento utilizadas no projeto de acordo com o PMBOK: Integração, Escopo, Tempo, Qualidade, Recursos Humanos, Comunicações e Riscos, visando sua integração com os grupos de processos, segundo a sua metodologia. No entanto, a área de conhecimento é definida por seus requisitos de conhecimentos e são descritas em termos dos processos que compõem suas práticas, entradas, saídas, ferramentas e 26 técnicas pertencentes as seguintes áreas de conhecimento: integração, escopo, cronograma, custos, qualidade, recursos, comunicação, riscos, aquisições e partes interessadas. (PMI, 2008).

4.2.1 Gestão do Escopo

O gerenciamento de escopo inclui processos e trabalho fundamental para iniciar, executar e finalizar o projeto. Nesta seção são apresentados a estrutura analítica do projeto e seu respectivo dicionário.

4.2.1.1 Estrutura Analítica do Projeto (EAP)

A EAP - em inglês *Work Breakdown Structure* (WBS) - consiste na representação hierárquica de todo o trabalho do projeto com a decomposição de seus resultados principais.

4.2.1.2 Dicionário EAP

O dicionário da EAP é a descrição detalhada de cada um dos componentes da EAP. Sendo o objetivo auxiliar tanto a equipe de desenvolvimento do projeto, quanto às demais partes interessadas na compreensão de cada um de seus componentes, o dicionário da EAP encontra-se junto ao cronograma no Apêndice R.

4.2.2 Gestão do Tempo

O gerenciamento de tempo inclui processos necessários que permitem as entregas do projeto dentro dos prazos estabelecidos. Esses processos permitem planejar o gerenciamento do cronograma, definir e sequenciar as atividades necessárias, estimar os recursos e durações das atividades, desenvolver e controlar o cronograma planejado.

4.2.2.1 Lista de Atividades, Diagrama de Rede e Cronograma

A lista de atividades refere-se a um conjunto de atividades a ser realizado para a construção do projeto e o diagrama de rede apresenta graficamente as dependências/relações entre elas. O cronograma de atividades do projeto é resultado da análise dos recursos necessários, das restrições do projeto, das durações e sequências das atividades, conforme Apêndice R.

4.2.2.2 Quadro com Resumo de Esforços

Nesta seção apresenta-se o resumo dos esforços para desenvolvimento do EduQ *Student Life*, com a previsão *ad-hoc*, estimativa do dicionário da EAP e o esforço real gasto com cada fase do escopo.

	Fase 1	Fase 2	Fase 3	Fase 4
Estimativa <i>ad-hoc</i>	180h/180h	200h/380h	250h/630h	200h/830h

Continua

Dicionário EAP	156h/156h	178h/334h	270h/604h	185h/789h
Efetivo realizado	95h/95h	180h/275h	240h/515h	160h/675h

QUADRO 2 - Esforços planejados e realizados por fase

Durante o planejamento da Fase 1, foi estimado um esforço com folga, pois os membros da equipe não possuíam muita experiência com relação ao planejamento das atividades durante o desenvolvimento do projeto. O que se pode observar é que ao final da primeira fase, o efetivo realizado ficou bem abaixo do planejado, isso ocorreu porque as atividades foram planejadas com muitas horas e, na prática, o tempo gasto ficou abaixo do esperado.

Durante o planejamento da Fase 2, foi estimado um esforço *ad-hoc* de 200 horas, quando devido à quantidade de tarefas relacionadas à modelagem, a equipe supôs que seriam gastas muitas horas. Porém, durante o desenvolvimento das atividades, a equipe constatou que a complexidade era mais baixa do que a esperada, com isso, foi gasto um tempo menor. Desta forma, o tempo planejado ficou maior do que o tempo real gasto no desenvolvimento das atividades da Fase 2.

Durante o planejamento da Fase 3 foi estimado um esforço *ad-hoc* de 250 horas, por conta do desenvolvimento do EduQ App e EduQ *Web Service*, além do tempo utilizado com testes. Porém, o tempo realmente gasto foi menor do que o esperado por conta da facilidade que o time obteve para desenvolver a interface do usuário no aplicativo Android, por esse motivo, a pequena folga de tempo na Fase 3.

No planejamento da Fase 4 foi estimado um esforço *ad-hoc* de 200 horas visto que a documentação a se fazer nessa fase é menor do que nas fases anteriores, mas o desenvolvimento, principalmente do aplicativo móvel foi maior do que nas fases anteriores. Assim, o tempo efetivo gasto na Fase 4 foi de 160 horas.

4.2.3 Gestão da Integração

O PMI (2008, p. 7) diz que o “gerenciamento da integração do projeto assegura que os diversos elementos do projeto sejam adequadamente coordenados. Envolve a tomada de decisão ligada diretamente aos objetivos do projeto e execução do plano do projeto”.

Nesta seção é apresentado como acontece a gestão da integração do projeto.

4.2.3.1 Monitoramento

O processo de monitoramento e controle consiste em acompanhar, organizar e analisar o desempenho do projeto com o objetivo de identificar fatores que necessitam de correções. O principal benefício deste processo é a medição e análise do desempenho do projeto a cada fase implementada, a fim de identificar as variações no plano de gerenciamento do projeto. Para acompanhamento das atividades é utilizado o *Microsoft Project* e também a ferramenta *Asana Web* que oferece um quadro de atividades, como pode ser visto na Figura 6.

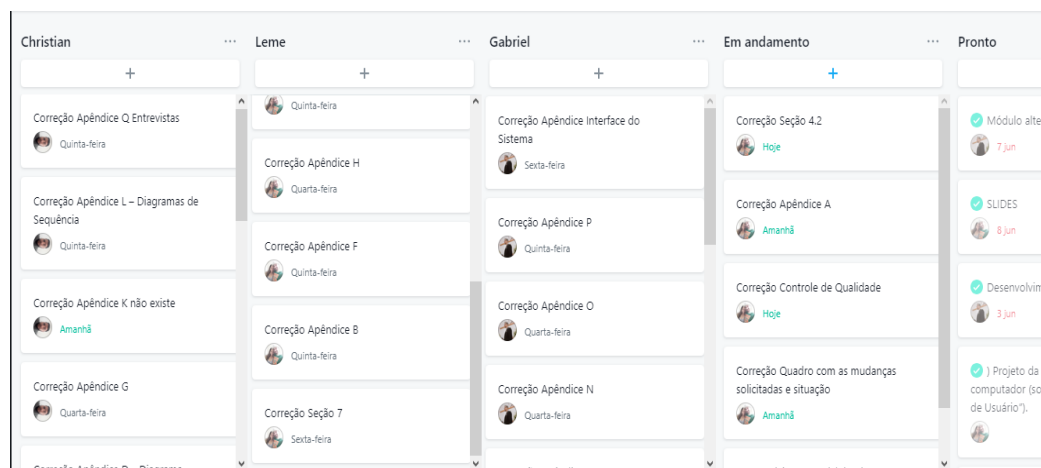


FIGURA 6 - Quadro de atividades no Asana.

O monitoramento é feito de forma diária, sempre considerando as atividades do quadro do Asana, com sua data final e inicial, o gerente de projetos pergunta qual o status da atividade programada e se há algum risco no desenvolvimento da mesma.

O relatório de desempenho encontra-se no Apêndice E.

4.2.3.2 Controle de Configuração

O gerente de configurações, Felipe Leme, é responsável por criar e manter um repositório padrão de armazenamento dos artefatos de documentação, banco de dados e componentes de *software* do projeto.


O repositório está organizado com uma pasta principal, onde dentro dela há subpastas que têm a identificação da versão de cada artefato já produzido. Sendo assim, o desenvolvedor poderá retirar apenas a versão que ele tem permissão para realizar alterações (controle realizado por definições de permissões aos componentes da equipe).


Cada necessidade de mudança no artefato da entrega deverá ser solicitada por um membro da equipe por meio de e-mail e utilizando o modelo disponível no Apêndice T. A mudança para ser atendida deve ser previamente aprovada pelos demais membros da equipe.

Para controle de versão do código-fonte utiliza-se o BitBucket, conforme mostra a Figura 7. Já o controle da documentação é feito com os recursos do Google Docs, conforme mostra a Figura 8.

Gabriel Rodrigo Martins da Silva

EduQWebApp Clonar ...

 master ▼ Filtrar arquivos

 /








Nome	Tamanho	Último commit	Mensagem
 .mvn/wrapper		2019-05-01	Initial commit
 src		2019-06-14	Correções diversas e algumas implementaçõe...
 .gitignore	303 B	2019-05-01	Initial commit
 mvnw	8.9 KB	2019-05-01	Initial commit
 mvnw.cmd	5.67 KB	2019-05-01	Initial commit
 pom.xml	2.82 KB	2019-05-29	Implementação do Interceptor para Escola

FIGURA 7 - Controle de versões BitBucket.

Shared with me > PFC EduQ > Entregas 





Name	Owner	Last modified	↓	File size
 Correção da entrega 2.	Carlos Christian	Jun 17, 2019	Carlos Christian	—
 Entrega 2	Carlos Christian	May 27, 2019	Carlos Christian	—
 Correção da entrega 1.	Carlos Christian	May 2, 2019	Carlos Christian	—
 Entrega 1	Carlos Christian	Apr 15, 2019	Carlos Christian	—

FIGURA 8 - Repositório da documentação.

4.2.3.2 Controle das Mudanças no Escopo

As solicitações de mudanças no escopo podem ser requisitadas por todas as partes interessadas do projeto. Essas solicitações são documentadas e avaliadas pelos membros da equipe de desenvolvimento, para que seja possível analisar essas informações, a fim de planejar, definir o escopo e recursos necessários para desenvolvimento das alterações aprovadas. Após a reunião de definição dos recursos essenciais, o gerente de projetos autoriza os desenvolvedores o acesso ao repositório correto para que comecem a execução de suas tarefas. Um exemplo de requisição de mudança está no Apêndice T.

Por fim, o gerente de projetos realiza todo controle e monitoramento das atividades para garantir que os prazos serão cumpridos de acordo com o seu planejamento.

4.2.4 Gestão da Qualidade

O gerente da qualidade, Felipe Leme, é responsável pelos processos de controle e garantia da qualidade do projeto. Os itens de avaliação são definidos de acordo com os aspectos e fatores relevantes aos procedimentos de codificação, comunicação e documentação do projeto.

Com o objetivo de manter a qualidade dos artefatos produzidos serão utilizadas ferramentas e normas, sendo elas: diretrizes de elaboração de documentos científicos da FAI, convenção de códigos Java e planilha contendo listas de verificação.

O controle de qualidade acontece por meio de revisão por outro membro da equipe de cada artefato desenvolvido, sendo que para cada ponto que o avaliador encontrar é feito um comentário e enviado por meio de e-mail para o responsável pelo artefato que deve analisar o comentário e tomar as providências necessárias.

4.2.5 Gestão dos Riscos

A gestão de riscos inclui os processos de identificação, análise, planejamento de respostas e controle dos riscos. O objetivo do gerenciamento é aumentar a probabilidade e impacto dos eventos positivos e reduzir os eventos que podem gerar impactos negativos no projeto.

A planilha dos riscos negativos encontra-se no Apêndice D.

5 ESPECIFICAÇÃO E ANÁLISE DE REQUISITOS

Este capítulo descreve explicitamente as funcionalidades e serviços do sistema de *software EduQ Student Life*, seus módulos e suas classificações em essenciais, importantes e desejáveis.

Além da descrição dos requisitos, este capítulo também traz os diagramas de casos de uso, modelo conceitual dos dados, modelo inicial da interface de usuário e as estimativas de esforço.

5.1 DESCRIÇÃO DE REQUISITOS

O levantamento de requisitos é a fase que se identifica o que o cliente deseja e o que espera em relação ao sistema. Um requisito é uma funcionalidade que o sistema deverá executar (BEZERRA, 2002). Esses requisitos podem ser levantados por entrevistas, questionários, observação do funcionamento do sistema atual, dentre outros.

5.1.1 Requisitos Funcionais

Os requisitos funcionais são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir a entradas específicas e de como o sistema deve se comportar em determinadas situações (SOMMERVILLE, 2011).

Neste projeto, os requisitos estão classificados da seguinte forma:

- a) Essencial: requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são imprescindíveis para o correto funcionamento do sistema;
- b) Importante: requisito sem o qual o sistema entra em funcionamento, porém de forma não satisfatória. Requisitos importantes devem ser implementados o mais rápido possível, mas, se não forem, parte do sistema poderá ser implementada mesmo assim;
- c) Desejável: requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele.

Requisitos desejáveis são requisitos que podem ser implementados por último, sem que comprometam o funcionamento do sistema.

Componente Aplicativo Web – EduQWeb

Neste componente, chamado de EduQWeb, estarão todas as funcionalidades que atenderão aos funcionários da instituição de Ensino Fundamental I.

RF01 – Cadastrar Escola

Este requisito refere-se ao cadastro de uma escola no sistema EduQWeb.

Para o cadastro serão necessários os dados: (1) código do MEC para a escola, (2) nome, (3) e-mail, (4) senha, (5) telefone, (6) CEP, (7) endereço, (8) município, (9) UF, (10) diretor e (11) imagem com logo da escola.

Ao finalizar o cadastro, o usuário poderá efetuar a autenticação no sistema (RF02) utilizando o código do MEC e a senha cadastrada.

Prioridade: essencial.

RF02 – Efetuar Autenticação da Escola

Este requisito refere-se à possibilidade de uma escola realizar a autenticação no sistema EduQ.

Para realizá-la serão necessários os dados: (1) código do MEC da escola e (2) senha previamente cadastrados (RF01).

Por meio desses dados, o usuário cadastrado como “Escola” terá acesso ao seu painel de controle onde poderá, por exemplo, cadastrar professores, classes e alunos.

Prioridade: essencial.

RF03 – Cadastrar Professor

Este requisito refere-se ao cadastro de um professor que atua como docente na escola.

Para o cadastro de um professor serão necessários os dados: (1) CPF, (2) nome, (3) e-mail, (4) senha, (5) telefone, (6) código gerado pela escola (RF 04), (7) data de nascimento, (8) endereço, (9) município e (10) UF.

Uma vez que o usuário de um professor seja criado por uma escola, esse poderá se autenticar e ter acesso as suas respectivas funções dentro do sistema, utilizando o código gerado pela escola e a senha informada (RF 05).

Prioridade: essencial.

RF04 – Gerar Código de Acesso para o Professor

Este requisito refere-se à funcionalidade de gerar um código de acesso para a autenticação do professor no sistema.

Para gerar o código serão necessários os dados: (1) código do MEC da escola e (2) CPF do professor.

O sistema deve gerar automaticamente um código de acesso numérico contendo 8 dígitos, que seja único para cada professor cadastrado na escola.

Prioridade: essencial.

RF05 – Efetuar Autenticação do Professor

Este requisito refere-se à possibilidade de um professor realizar sua autenticação no sistema EduQ.

Para realizar a autenticação serão necessários os dados: (1) código gerado pela escola (RF04) e (2) senha, previamente, cadastrados (RF03).

Por meio desses dados, o usuário cadastrado como professor terá acesso ao seu painel de controle onde poderá, por exemplo, realizar uma chamada (RF18) e enviar avisos aos responsáveis (RF20).

Prioridade: essencial.

RF06 – Alterar Senha

Este requisito refere-se à possibilidade de um usuário alterar sua senha no sistema caso seja necessário.

Para alterar a senha será necessário: (1) estar autenticado no sistema (RF02 e RF05), (2) senha previamente cadastrada para ser alterada e (3) nova senha (RNF09).

O usuário poderá alterar sua senha quando achar necessário, sendo necessário apenas estar autenticado no sistema e possuir sua antiga senha.

Prioridade: essencial.

RF07 – Redefinir Senha

Este requisito refere-se à possibilidade de o usuário restaurar sua senha no sistema caso venha a esquecer a antiga.

Para restaurar a senha: (1) código de acesso ao sistema (RF01 e RF04).

Caso o usuário esqueça sua senha, ele poderá solicitar a redefinição dela, sendo que o próprio sistema irá gerar uma senha alfanumérica aleatória (RF08) e irá enviar ao e-mail previamente vinculado ao usuário (RF09).

Prioridade: essencial.

RF08 – Gerar Senha Aleatória

Este requisito refere-se à possibilidade de o sistema gerar uma senha alfanumérica aleatória para um usuário que tenha solicitado a redefinição de sua senha de acesso.

Para restaurar a senha: (1) código de acesso ao sistema (RF01 e RF04).

O sistema deverá gerar uma sequência alfanumérica (RNF09) que corresponderá a uma senha temporária para que o usuário possa se autenticar no sistema (RF02 e RF05) e fazer a alteração de sua senha (RF06).

Prioridade: essencial.

RF09 – Enviar E-mail

Este requisito refere-se à possibilidade de o sistema enviar um e-mail assim que necessário ao usuário com informações relevantes para seu uso.

Para enviar um e-mail: (1) E-mail previamente cadastrado de um usuário (RF01 e RF03).

O sistema deverá ser capaz de enviar um e-mail ao usuário contendo informações úteis a seu uso, como a confirmação de seu cadastro (RF01 e RF03) ou a nova senha redefinida.

Prioridade: essencial.

RF10 – Efetuar Saída

Este requisito refere-se à possibilidade de um usuário se desconectar do sistema, independente se for um professor ou uma escola.

Para efetuar a saída: (1) estar autenticado no sistema (RF02 e RF05) com código de acesso e senha previamente cadastrados (RF01 e RF03).

O usuário poderá sair a qualquer momento de seu painel de controle (RF11).

Prioridade: essencial.

RF11 – Administrar Conta

Este requisito refere-se à possibilidade de um usuário, seja ele um professor ou uma escola, administrar sua conta e/ou dados de forma centralizada em um painel de controle.

Para administrar a conta: (1) estar autenticado no sistema (RF02 e RF05) com código de acesso e senha previamente cadastrados (RF01 e RF03).

O usuário poderá a qualquer momento retornar ao seu painel de controle para conferir seus dados de forma que possa alterar seus dados (RF12) a menos que esteja no meio de algum tipo de transação no sistema. Neste painel de controle, o usuário terá acesso a todas as suas atribuições de acordo com o tipo de cadastro que tiver.

Prioridade: essencial.

RF12 – Manter Perfil

Este requisito refere-se à edição de um perfil cadastrado.

Para editar um perfil: (1) estar autenticado no sistema (RF02 e RF05) com código de acesso e senha previamente cadastrados (RF01 e RF03).

Uma vez autenticado, o usuário poderá editar seus dados pessoais, bem como complementar, caso falte algum dado no cadastro.

Prioridade: essencial.

RF13 – Cadastrar Dados do Aluno

Este requisito refere-se à possibilidade de cadastrar um aluno no sistema EduQ que esteja matriculado na escola de referência.

Para cadastrar um aluno: (1) o usuário deverá estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01). Serão necessários os seguintes dados: (2) CPF, (3) nome, (4) matrícula, (5) data da matrícula, (6) CPF do responsável, (7) código gerado pela escola (RF28), (8) data de nascimento, (9) nome da mãe, (10) nome do pai, (11) responsável e (12) situação do cadastro.

Um usuário autenticado como “escola” terá a permissão de cadastrar, excluir (RF14) e alterar o cadastro de um aluno (RF15).

Um aluno cadastrado terá uma situação, sendo ela ativa ou inativa. Na situação ativa, o aluno está regularmente matriculado e em curso em uma classe, de forma que nesta situação o mesmo não poderá ser excluído. Já a situação inativa indica que o aluno já não estuda mais na escola e assim pode ter seu cadastro removido do sistema.

Prioridade: essencial.

RF14 – Excluir Dados do Aluno

Este requisito refere-se à possibilidade de excluir um aluno do sistema EduQWeb por parte da escola.

Para excluir um aluno: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01) e (2) matrícula do aluno.

Um usuário autenticado como “escola” terá a permissão para excluir um aluno, desde que o aluno não esteja vinculado a uma classe (RF16).

Prioridade: essencial.

RF15 – Alterar Cadastro de Aluno

Este requisito refere-se à possibilidade de alterar o cadastro um aluno no sistema EduQWeb por parte de uma escola.

Para alterar os dados de um aluno será preciso: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01), (2) matrícula do aluno.

Um usuário autenticado como uma escola terá a permissão para alterar o cadastro de um aluno, desde que o aluno não esteja como inativo em relação a uma escola.

Prioridade: desejável.

RF16 – Criar Classe

Este requisito refere-se à possibilidade de criar uma classe dentro do registro de uma escola no sistema EduQWeb.

Para criar turma: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01), (2) ano da classe, (3) código da classe, (4) ano letivo da classe, (5) código do professor responsável pela classe (RF04), (6) turno e (7) descrição.

Um usuário autenticado como escola terá a possibilidade de criar uma classe e atribuí-la a um professor. O professor irá realizar a chamada diária desta classe (RF18), bem como

enviar avisos aos responsáveis dos alunos da classe (RF20). O professor também irá lançar os conceitos de cada aluno no seu respectivo boletim (RF21).

Prioridade: essencial.

RF17 – Adicionar Aluno para Classe

Este requisito refere-se à possibilidade de adicionar um aluno a uma respectiva classe no sistema EduQWeb.

Para adicionar um aluno a uma classe: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01), (2) ano da classe, (3) código da classe e (4) matrícula do aluno.

Um usuário autenticado como escola terá a possibilidade de adicionar um aluno a uma classe, desde que o aluno não esteja vinculado a outra classe em uma escola diferente. Caso o aluno esteja vinculado a outra classe na mesma escola, quando o aluno for adicionado a outra classe, ele irá ser excluído da classe anterior.

Prioridade: essencial.

RF18 – Realizar Chamada Diária (Controle de Frequência)

Este requisito refere-se à possibilidade de um professor realizar a chamada diária de uma classe em específico.

Para realizar a chamada: (1) estar autenticado no sistema (RF05) com código de acesso e senha previamente cadastrados (RF03), (2) ano da classe, (3) código da classe e (4) data da chamada.

Um professor terá a possibilidade de realizar uma nova chamada diária de uma respectiva classe, tendo em vista que ao realizar esta chamada irá adicionar as faltas individuais de cada aluno. Caso seja necessário alterar a frequência à aula, ele poderá realizar normalmente (RF19).

Prioridade: essencial.

RF19 – Editar Chamada Diária

Este requisito refere-se à possibilidade de um professor editar o lançamento de uma chamada, alterando a presença ou falta de um aluno caso seja necessário.

Para editar uma chamada: (1) estar autenticado no sistema (RF05) com código de acesso e senha previamente cadastrados (RF03), (2) ano da classe, (3) código da classe e (4) data da chamada.

Uma vez que se faz necessário editar uma chamada diária, o professor poderá alterar a presença ou falta de um aluno caso seja necessário.

Prioridade: essencial.

RF20 – Enviar Aviso

Este requisito refere-se à possibilidade de um professor enviar avisos aos responsáveis pelos alunos de uma classe.

Para enviar um aviso: (1) estar autenticado no sistema (RF05) com código de acesso e senha previamente cadastrados (RF03), (2) ano da classe, (3) código da classe, (4) data do aviso, (5) tipo do aviso e (6) conteúdo.

O professor poderá enviar avisos aos responsáveis pelos alunos da classe de forma que estes avisos podem ser datas de provas com o seu respectivo valor e assunto, data de apresentação de trabalhos, viagens e até mesmo uma notificação de reunião de pais/responsável por parte da escola.

Prioridade: essencial.

RF21 – Gerar Boletim Escolar

Este requisito refere-se à possibilidade de o sistema gerar um boletim individual para cada aluno cadastrado no sistema a cada ano em que ele continue ativo no EduQWeb.

Para gerar o boletim: (1) matrícula do aluno, (2) código da classe, (3) turno, (4) faltas, (5) observação, (6) conceito, (7) situação, (8) bimestre, (9) data e (10) componente curricular.

O sistema deve gerar o boletim individual para cada aluno e à medida que as notas dos componentes curriculares (RF23) forem sendo lançadas, o sistema irá automaticamente acrescentar/incrementar as notas no boletim do aluno. O sistema deve também atualizar automaticamente as faltas do aluno à medida que elas forem lançadas.

Prioridade: essencial.

RF22 – Editar Boletim Escolar

Este requisito refere-se à possibilidade de uma escola editar no boletim a situação, nota e/ou matrícula do aluno vinculado.

Para editar o boletim: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01), (2) ano da classe, (3) código da classe, (4) matrícula do aluno, (5) código do componente curricular e (6) Identificação do boletim.

O sistema deve permitir que a escola venha a editar os dados que constam no boletim, a fim de que possam corrigir algum lançamento.

Prioridade: essencial.

RF23 – Cadastrar Componente Curricular

Este requisito refere-se à possibilidade de uma escola cadastrar os componentes curriculares que os professores irão trabalhar com os alunos.

Para cadastrar uma disciplina: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01), (2) nome do componente, (3) carga horária semanal, (4) tipo de avaliação e (5) código do componente curricular.

Uma vez autenticada no sistema, a escola poderá cadastrar os componentes curriculares que deverão ser trabalhadas pelos professores de acordo com a BNCC.

Prioridade: essencial.

RF24 – Editar Componente Curricular

Este requisito refere-se à possibilidade de uma escola alterar um componente curricular anteriormente cadastrado.

Para editar uma disciplina: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01) e (2) nome do componente curricular.

Uma vez autenticada, a escola poderá alterar os dados dos componentes curriculares que deverão ser trabalhados pelos professores com os alunos.

Prioridade: desejável.

RF25 – Obter Ajuda

Este requisito refere-se à possibilidade de um usuário, seja ele professor ou uma escola, obter ajuda sobre a operação do sistema a qualquer momento.

Para obter ajuda: (1) estar autenticado no sistema (RF02 e RF05) com código de acesso e senha previamente cadastrados (RF01 e RF03).

O usuário pode a qualquer momento acessar seu painel de controle para que tenha acesso aos tópicos de ajuda funcional, bem como enviar uma dúvida diretamente aos administradores do sistema (RF27).

Prioridade: desejável.

RF26 – Notificar Administrador

Esse requisito refere-se à possibilidade de um usuário se comunicar com os administradores do sistema a qualquer momento.

Para se comunicar com os administradores: (1) estar autenticado no sistema (RF02 e RF05) com código de acesso e senha previamente cadastrados (RF01 e RF03) e (2) acessar a opção de obter ajuda no painel de controle (RF25).

O usuário, após acessar a opção de ajuda no painel de controle, poderá enviar sua dúvida, sugestão ou crítica diretamente aos administradores do sistema e receberá sua resposta via e-mail.

Prioridade: desejável.

RF27 – Gerar Código do Aluno para Acesso do Responsável

Este requisito refere-se à funcionalidade de gerar um código de registro para o acompanhamento do aluno por seu responsável no sistema EduQ.

Para gerar o código serão necessários os dados: (1) código do MEC da Escola, (2) CPF do aluno.

O sistema deverá gerar automaticamente um código de acesso numérico contendo 14 dígitos, que seja único para cada aluno cadastrado na escola.

Prioridade: essencial.

RF28 - Cadastrar Calendário do Ano Letivo

Este requisito refere-se ao cadastro do calendário letivo do ano, a fim de que o período de cada bimestre seja definido, assim como datas de recesso escolar, feriados, dentre outros. Esse calendário será consultado para o controle de frequências às aulas.

Prioridade: essencial.

RF29 – Cadastrar Eventos Escolares

Este requisito refere-se à possibilidade de uma escola cadastrar os eventos escolares em um calendário interno.

Para cadastrar um evento: (1) estar autenticado no sistema (RF02) com código de acesso e senha previamente cadastrados (RF01).

Uma vez autenticada, a escola poderá cadastrar os eventos necessários que serão disponibilizados aos professores em seus calendários.

Prioridade: desejável.

Aplicativo *Mobile* - EduQApp

Neste componente, chamado de EduQApp, estarão todas as funcionalidades que atenderão aos responsáveis pelos alunos vinculados a alguma das Instituições de Ensino Fundamental I, cadastradas no sistema EduQ.

RF30 – Cadastrar Conta de Acesso do Responsável com E-mail

Este requisito refere-se à possibilidade de um responsável se cadastrar no sistema através do EduQApp.

Para cadastrar um responsável: (1) e-mail e (2) senha.

Um usuário poderá iniciar seu cadastro no EduQApp como responsável apenas informando um e-mail válido e uma senha. O sistema irá enviar um e-mail de verificação (RF29) para o endereço informado a fim de validar os dados informados pelo usuário.

Prioridade: essencial.

RF31 – Enviar E-mail de Confirmação

Esse requisito refere-se ao envio de uma mensagem de validação de cadastro ao e-mail informado no EduQApp no momento do cadastro a fim de validar a informação.

Para enviar o e-mail de confirmação: (1) e-mail previamente cadastrado (RF28).

O sistema irá enviar um e-mail de confirmação ao endereço informado contendo um código de confirmação que deverá ser informado no aplicativo para que o cadastro possa ser completado.

Prioridade: essencial.

RF32 – Atualizar Cadastro do Responsável

Este requisito refere-se à possibilidade de atualização do cadastro de um responsável no aplicativo EduQApp.

Para completar o cadastro: (1) e-mail previamente cadastrado (RF28), (2) nome, (3) telefone, (4) CEP, (5) endereço e (6) município.

Uma vez que o usuário tenha validado o seu *e-mail*, será necessário que ele complete o seu cadastro informando os dados descritos e assim ser liberado para acompanhar um aluno.

Prioridade: essencial.

RF33 – Manter-se Conectado

O sistema deverá oferecer a funcionalidade de que um responsável possa se manter conectado ao aplicativo mesmo quando o mesmo for finalizado.

Para se manter conectado: (1) e-mail previamente cadastrado (RF28) e (2) selecionar a opção relacionada.

Uma vez que usuário preencha os dados para efetuar sua autenticação, o sistema deverá oferecer uma opção para que ele possa se manter conectado ao EduQApp.

Prioridade: essencial.

RF34 – Efetuar Autenticação do Responsável

Este requisito refere-se à possibilidade de um responsável efetuar sua autenticação no EduQApp.

Para realizar a autenticação serão necessários os dados: (1) e-mail e senha previamente cadastrados (RF28).

O usuário cadastrado como responsável terá acesso ao seu aplicativo onde poderá, por exemplo, acompanhar o boletim do aluno (RF37) e enviar avisos aos professores (RF36).

Prioridade: essencial.

RF35 – Cadastrar com uma Conta do Google

Este requisito refere-se à possibilidade de que o usuário possa se cadastrar e fazer sua autenticação com os dados de sua conta no Google.

Para se cadastrar com o Google: (1) o usuário deve possuir uma conta ativa no Google.

O usuário pode utilizar de sua conta no Google para eliminar alguns passos na criação de seu perfil, de forma que poderá completá-lo posteriormente dentro do sistema. Uma vez que o usuário utilize esse método, automaticamente, o usuário deverá permanecer conectado.

Prioridade: essencial.

RF36 – Efetuar Saída

Este requisito refere-se à possibilidade um responsável se desconectar do EduQApp. Para efetuar a saída: (1) estar autenticado no sistema (RF33 e RF34)

O usuário poderá sair a qualquer momento de sua página de perfil (RF36).

Prioridade: essencial.

RF37 – Administrar Conta do Responsável

Este requisito refere-se à possibilidade de um responsável, administrar sua conta e/ou dados de forma centralizada por meio de um painel de controle.

Para administrar a conta: (1) estar autenticado no sistema (RF33 e RF34).

O usuário poderá a qualquer momento retornar a sua tela de perfil para conferir seus dados de forma que possa alterar seus dados (RF37) a menos que esteja no meio de algum tipo de transação no sistema. Neste painel de controle, o usuário terá acesso a todas as suas atribuições de acordo com o tipo de cadastro que tiver.

Prioridade: essencial.

RF38 – Manter Perfil do Responsável

Este requisito refere-se à edição de um perfil cadastrado.

Para editar um perfil: (1) estar autenticado no sistema (RF33 e RF34).

Uma vez autenticado, o usuário poderá editar seus dados pessoais, bem como complementar, caso falte algum dado no cadastro.

Prioridade: essencial.

RF39 – Enviar Solicitação

Este requisito refere-se à possibilidade de que o responsável possa enviar solicitações e avisos à escola.

Para enviar solicitações: (1) estar autenticado no sistema (RF33 e RF34), (2) tipo da solicitação, (3) conteúdo e (4) código do aluno.

Uma vez autenticado, o responsável deverá conseguir enviar solicitações e avisos aos professores, informando o tipo da solicitação (ou aviso).

Prioridade: essencial.

RF40 – Acompanhar Atividades do Aluno

Este requisito refere-se à possibilidade de um responsável acompanhar a vida estudantil de um estudante.

Para acompanhar: (1) estar autenticado no sistema (RF33 e RF34) e (2) código do aluno gerado pela escola (RF27).

Uma vez autenticado, o aplicativo deverá disponibilizar uma opção por meio da qual o responsável irá informar o código que foi gerado pela escola para o aluno, e assim terá acesso a suas informações e aos avisos encaminhados pela escola.

Prioridade: essencial.

Componente *Web Service* Firebase

Nestes componentes estarão todas as funcionalidades que atenderão aos demais componentes do sistema.

RF41 – Sincronizar Avisos

Este requisito refere-se à possibilidade de que um *Web Service* desenvolvido no Firebase, sincronize os dados entre os componentes *Web* e *Mobile*.

Para sincronizar: (1) monitorar envio de avisos ou requisições de ambos os componentes (RF41).

O *Web Service* de sincronismo deve fazer o monitoramento periódico do envio de solicitações e avisos por parte dos responsáveis pelos alunos, bem como de lançamentos realizados pelos professores ou a escola e assim sincronizar as notificações entre as partes.

Prioridade: essencial.

RF42 – Monitorar Lançamentos

Este requisito refere-se à possibilidade de que um *Web Service* desenvolvido no Firebase, monitore os lançamentos feitos no sistema.

Para sincronizar: (1) detectar envios de solicitações e avisos entre os componentes.

O *Web Service* de monitoramento deverá receber as solicitações e avisos de ambas as partes e assim iniciar o *Web Service* de sincronismo (RF40).

Prioridade: essencial.

5.1.2 Requisitos Não Funcionais

Os requisitos não funcionais definem propriedades relacionadas ao sistema, como confiabilidade e ocupação de espaço em disco, definindo restrições sobre a implementação do sistema. Desta forma, ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo (SOMMERVILLE, 2011).

RNF01 – Usabilidade

O sistema deverá apresentar interfaces de usuário simples e intuitivas para o usuário. Os usuários devem ser capazes de compreender a linguagem visual dos componentes *Web* e *mobile* e utilizarem suas funções de maneira fácil, intuitiva e com alto nível de satisfação.

Prioridade: essencial.

RNF02 – Disponibilidade

Por se tratar de um sistema com componentes *web* e *mobile* se faz necessário um sistema altamente disponível, algo em torno de 99,4% do dia. O sistema deverá ser hospedado em um servidor de aplicações Java, disponível pela Internet, a fim de que os usuários possam usufruir das suas funcionalidades a qualquer momento e localidade por meio de um dispositivo com acesso à Internet.

Prioridade: essencial.

RNF03 – Data marco do projeto

O projeto durante o período de desenvolvimento terá quatro entregas, respectivamente nas seguintes datas: 06 de abril de 2019, 01 de junho de 2019, 07 de setembro de 2019 e 02 de novembro de 2019.

Prioridade: essencial.

RNF04 - Linguagem de Programação e Sistema de Gerenciamento de Banco de Dados (SGBD)

O sistema será desenvolvido em linguagem de programação Java em suas plataformas *web* e *mobile*, também será utilizado o banco de dados PostgreSQL v.9 e FireBase v.16.0.8.

Prioridade: essencial.

RNF05 - Privacidade

O sistema garantirá privacidade dos dados armazenadas no banco de dados, de modo que não permitirá aos usuários terem acesso aos dados sigilosos sem a autorização. O controle de privacidade ocorrerá via atribuição de permissões que serão configuradas de acordo com o perfil do usuário.

Prioridade: essencial.

RNF06 – Servidor Web

O sistema utilizará o *container web Apache Tomcat v.8*. Esse servidor deverá apresentar resultados satisfatórios em sua interoperabilidade com o sistema EduQ.

Prioridade: essencial.

RNF07 – Layout Responsivo

O sistema será elaborado para ser utilizada em diferentes dispositivos que possuam uma conexão com a Internet. Sejam eles: computadores, *tablets* ou *smartphones* de todos os tamanhos e formas e deve possuir um *layout* ajustável para melhor visualização em suas telas de variados tamanhos.

Prioridade: essencial.

RNF08 – Interoperabilidade com Navegadores

O sistema fará uso de navegadores para o acesso de usuário externos, como o Microsoft Edge (a partir da versão 25.10586.0.0), Google Chrome (a partir da versão 74.0.3729.169), Mozilla Firefox (a partir da versão 61.0) e Safari (a partir da versão 11.1.2).

Prioridade: essencial.

RNF09 – Política para Formação de Senhas

As senhas que devem ser cadastradas no sistema devem ser formadas por no mínimo oito caracteres alfanuméricos. As senhas não podem ser iguais aos dados pessoais informados, por exemplo, nome e telefone.

Prioridade: essencial.

RNF10 – Sistema Operacional do Componente *Mobile*

O EduQApp deve ser desenvolvido para ser executado no sistema operacional Android, com sua versão mínima sendo a Lollipop 5.1.

Prioridade: essencial.

RNF11 – Protocolo Seguro para Transferência de Dados entre Componentes

O sistema EduQ deve fazer uso de uma camada de segurança na comunicação entre seus componentes, sendo ela a tecnologia SSL/TLS como complemento ao protocolo HTTPS.

Prioridade: essencial.

RNF12 – Interoperabilidade com o Google

O sistema EduQ deve trabalhar em conjunto com o servidor Google a fim de manter o funcionamento completo dos requisitos relacionados ao componente de *web service* Firebase e também ao cadastro de um responsável com uma conta do Google.

Prioridade: essencial.

5.2 ANÁLISE DOS REQUISITOS

Nesta seção apresentam-se os modelos de caso de uso, o modelo conceitual dos dados e da interface de usuário.

5.2.1 Modelo de Casos de Uso

Os diagramas de casos de uso estão no Apêndice B e as descrições dos fluxos de eventos estão no Apêndice K.

5.2.2 Modelo Conceitual dos Dados

O modelo conceitual dos dados de um sistema tem alto nível de abstração e mostra as relações entre as entidades que participam do domínio do problema. Para sua representação, neste projeto, utiliza-se o Diagrama de Entidade e Relacionamento que está no Apêndice C.

5.2.3 Modelo Inicial da Interface de Usuário

Esta seção mostra alguns protótipos das interfaces de usuário previstas para o EduQApp e EduQWeb. Uma lista completa das telas está disponível no Apêndice F.

5.3 MÉTRICAS PARA ESTIMATIVA DE ESFORÇO

Empregando as técnicas de Pontos de Função (PF) e Pontos por Casos de Uso (PCU) é calculado o tamanho do sistema e a partir desse é possível realizar uma estimativa de esforço para seu desenvolvimento.

Segundo Pressman (2006), a métrica de PF pode ser entendida como uma medida da funcionalidade fornecida pelo *software*, que pode ser usada para calcular o custo ou trabalho necessário do projeto, antecipar o número de erros que serão encontrados e pressupor o número recomendado de integrantes. Para estimar o esforço necessário para o desenvolvimento ainda na fase de levantamento de Casos de Uso e tendo como base a complexidade compreendida, Karner (1993) criou a métrica de PCU.

As estimativas de PF e PCU são apresentadas no Quadro 3. Para a métrica de PCU, foram realizadas as estimativas segundo os métodos de Schneider e Winters e de Karner. Todas as estimativas foram realizadas a partir da segunda fase do projeto e deve-se considerar que mudanças ocorreram entre elas, devido ao melhor entendimento do projeto pela equipe. As planilhas com detalhamento dos valores obtidos estão no Apêndice S.

Métrica	Fase 2 (Hh)	Fase 3 (Hh) (Reestimativa)	Fase 4 (Hh) (Reestimativa)
PCU - Karner	1665 Hh	1892 Hh	1893 Hh
PCU - Schneider e Winters	1665 Hh	1892 Hh	1893 Hh
Pontos por Função	-	5530 Hh	10044 Hh

QUADRO 3 - Estimativas de esforços de desenvolvimento

6 ARQUITETURA E PROJETO DO SISTEMA DE SOFTWARE

Este capítulo apresenta a arquitetura e a modelagem do sistema EduQ *Student Life*. Os diagramas expostos neste capítulo devem auxiliar para um melhor conhecimento do sistema quanto à sua estrutura, seu comportamento, dados sujeitos de armazenamento e sua visão física e de distribuição.

6.1 VISÃO ESTRUTURAL

Nesta seção apresentam-se o diagrama de classes, diagrama de pacotes, diagrama de classes o diagrama de objetos.

6.1.1 Diagrama de Pacotes

O diagrama de pacotes da *Unified Modeling Language* (UML) descreve os pacotes ou pedaços do sistema divididos em agrupamentos lógicos e mostra as dependências entre eles.

Para o projeto EduQ *Student Life* é utilizada a arquitetura de *software* Modelo – Visão - Controlador, da expressão em inglês *Model – View – Controller* (MVC). A camada de modelo (*model*) consiste na manipulação dos dados do sistema, nas regras de negócio, lógicas e funções. Já a camada de Visão (*view*) consiste na entrada e apresentação dos dados, ou seja, na interação direta com o usuário. Enquanto a camada Controlador (*controller*) faz a mediação entre as camadas de Visão e de Modelo.

Os diagramas de pacotes do EduqWeb e EduqApp encontram-se no Apêndice G.

6.1.2 Diagramas de Classes

Diagrama de classes é uma representação estática utilizada na área da programação para descrever a estrutura de um sistema, apresentando suas classes, atributos, operações e as relações entre os objetos. Os diagramas de classe do pacote *Web* está disponível no Apêndice H.

6.1.3 Diagramas de Objetos

Os diagramas de objetos fornecem uma captura instantânea das instâncias em um sistema e os relacionamentos entre as instâncias, mostrando os elementos de modelos em um diagrama de classe, podendo explorar o comportamento de um sistema em um determinado momento. Um diagrama de objetos para o caso de uso “Manter Cadastro Professor” encontra-se no Apêndice I.

6.2 VISÃO COMPORTAMENTAL

Nesta seção apresentam-se os diagramas que mostram as mensagens entre os objetos no decorrer do tempo para a realização de uma operação. Também apresenta o diagrama de visão geral de interação, cujo seu principal objetivo é mostrar uma visão geral do controle de fluxo das atividades entre as interações.

6.2.1 Projeto das Interações entre Objetos

O projeto das interações visa relacionar os diversos objetos do sistema no decorrer da realização de um ou mais casos de uso.

6.2.1.1 Diagramas de Sequência

Diagrama de Sequência é um dos modelos da UML usado para representar interações entre objetos de um cenário, realizadas por meio de operações ou métodos (procedimentos ou funções). Este diagrama é construído a partir do Diagrama de Casos de Uso.

Alguns diagramas de sequência para o caso de uso “Manter cadastro de professor” encontram-se no Apêndice J.

6.2.1.2 Diagrama de Visão Geral da Interação

Os diagramas de visão geral de interação fornecem um alto nível de abstração em um modelo. Eles são semelhantes a diagramas de atividades, mas também mostram a interação entre outros diagramas de interação. Um diagrama de visão geral de interação pode mostrar um fluxo de controle normal ou alternativo em um sistema. A notação do diagrama de visão

geral de interação é semelhante à notação para o diagrama de atividades e de sequência. Um diagrama de visão geral da interação encontra-se no Apêndice L.

6.2.2 Diagrama de Atividades

A UML inclui diversos subconjuntos de diagramas, incluindo diagramas estruturais, de interação e de comportamento. Diagramas de atividade, junto com diagramas de caso de uso e de máquina de estados, são considerados diagramas de comportamento porque descrevem o que é necessário acontecer no sistema sendo modelado.

Um diagrama de atividades que modela o caso de uso “Enviar uma requisição para um responsável” encontra-se no Apêndice M.

6.3 VISÃO DOS DADOS

Segundo Silberschatz (2006), um dos benefícios de um banco de dados é proporcionar ao usuário uma visão abstrata dos dados. Isto é, o sistema acaba por ocultar determinados detalhes sobre a forma de armazenamento e manutenção desses dados.

A Figura 9 apresenta as múltiplas visões de dados, pois um banco de dados deve permitir o acesso de diversos tipos de usuários a todo o seu conteúdo, é possível imaginar que cada usuário, ou grupo, tenham suas necessidades mais específicas. Sendo assim, é necessário que cada conjunto de usuário tenha a possibilidade de ter visões diferentes da base de dados e também nesse ponto assegurar privilégios de acessos a essas visões.

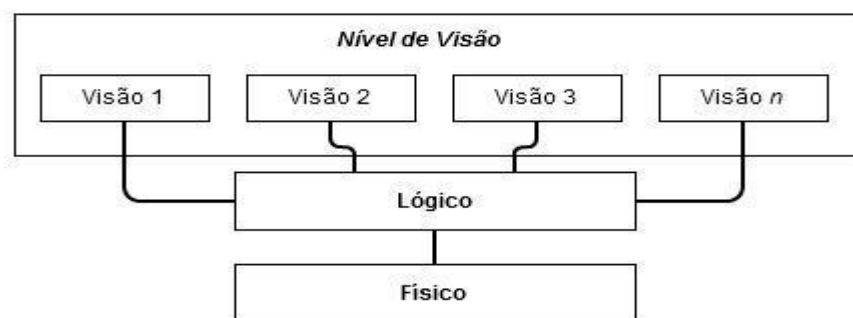


FIGURA 9 - Níveis de visão de um banco de dados.

Fonte: Silberschatz (2006).

6.3.1 Modelo Lógico

O modelo lógico também conhecido como modelo operacional é arquitetado a partir do modelo conceitual do projeto. Ele avalia as limitações da base de dados e implementa recursos como adequação de padrão e nomenclatura, definindo chaves primárias e estrangeiras das tabelas. O modelo lógico encontra-se no Apêndice N.

6.3.2 Dicionário de Dados do Modelo Lógico

O dicionário de dados consiste numa lista organizada de todos os elementos de dados que são pertinentes para o sistema. Ele consiste num ponto de referência de todos os elementos envolvidos na medida em que permite associar um significado a cada termo utilizado. O dicionário de dados encontra-se no Apêndice O.

6.3.3 Estrutura Física do Banco de Dados

Artefato	Referência
<i>Scripts DDL</i>	Vide Apêndice V.
<i>Scripts DML</i>	Vide Apêndice V.
<i>Views</i> , funções de gatilho e <i>store procedures</i>	Vide Apêndice V.
Tipos de usuários e permissões de acesso	Vide Apêndice V.

QUADRO 4 - Modelo físico do banco de dados

6.4 VISÃO FÍSICA

A visão física apresenta os componentes que compõem a arquitetura do sistema e como eles estão relacionados. Nesta seção é apresentado o diagrama de componentes para ilustrar a visão física do sistema, os *frameworks* adotados para o projeto, o *design patterns* aplicados, as convenções e guias para codificação e a análise de complexidade algorítmica.

6.4.1 Diagrama de Componentes

O diagrama de componentes é a representação dos componentes, suas *interfaces* e suas dependências, os objetivos principais esperados em um diagrama de componentes são:

- a. definir os pontos executáveis e reutilizáveis do sistema;
- b. mostrar a configuração do *software* por meio dos relacionamentos de dependência;

O diagrama de componentes do EduQ *Student Life* encontra-se no Apêndice U.

6.4.2 Frameworks Adotados

Um *framework* descreve a arquitetura de um sistema genérico, os tipos de objetos e as interações entre os mesmos. Ele pode ser visto como o esqueleto de um sistema que pode ser customizado conforme a necessidade em questão.

Para Johnson (1997, p.39–42), um *framework* pode facilitar a construção de novos componentes e fornecer uma *interface* padrão para os mesmos trocarem dados, manipularem erros e operações.

No desenvolvimento do EduQ *Student Life* empregam-se os seguintes *frameworks*:

- a. Spring Boot: *framework* que visa solucionar a complexidade da criação e gerenciamento de dependências de um projeto com Spring e Java, além de tratar de maneira coesa e eficiente a questão da configuração. Gerencia dependências de maneira automática e simplifica a execução do projeto em tempo de desenvolvimento e depuração. O Spring Boot também possui a funcionalidade de empacotamento da aplicação em um arquivo JAR(*Java Archive*) executável contendo todas as dependências necessárias;
- b. Hibernate: *framework* de persistência, responsável pelo mapeamento de objetos Java em tabelas do banco de dados, simplificando assim a persistência e leitura de dados;
- c. Bootstrap: *framework* para construção do *front-end* disponibilizado pelo Twitter. Provê facilidade no desenvolvimento de *front-end* visto que o desenvolvedor não precisa se preocupar em editar CSS(*Cascading Style Sheets*), apenas informar a qual classe aquele componente HTML(*Hypertext Markup Language*) pertence e o estilo do componente é automaticamente aplicado ao componente pelo Bootstrap.

6.4.3 Design Patterns Aplicados

Um *Design Pattern* é basicamente uma forma anteriormente implementada e testada que fornece, de maneira genérica, uma solução para problemas específicos.

Propósito	Padrões utilizados	Exemplos de classes e/ou métodos aplicados
Arquitetural	MVC	<i>Entity, Service e Controller</i>
Criacional	Singleton	<i>EntityManager</i>
Comportamental	Injeção de dependência Iterator	ProfessorController
Estrutural	<i>Data Access Object</i> (DAO)	ProfessorDAO

QUADRO 5 - Apresentação dos padrões utilizados

- a. **MVC:** é um padrão arquitetural que possui como premissa básica fornecer uma divisão trilateral de camadas: *model*, *view* e *controller*. O *controller* é responsável por fazer a conexão entre o usuário e as regras de negócio contidas no *model*, assim, é por meio dele que as solicitações e requisições da *view* são transmitidas ao *model* e retornadas à *view*. O *model* é responsável por conter todas as regras de negócio da aplicação e, além disto, as classes de persistência com o sistema gerenciador de banco de dados. A *view* é a camada na qual o usuário interage diretamente com o sistema;
- b. **Injeção de dependência:** é um padrão que trabalha baseado em abstrações, sejam elas classes abstratas ou interfaces, sendo dividido em três partes injeção por construtor (*constructor injection*); injeção por propriedade (*setter injection*) e injeção por interface (*interface injection*);
- c. **DAO:** é um padrão utilizado que representa o pacote que é composto pelas classes de persistência com o banco de dados. Essas classes são compostas de operações de inserção, leitura, alteração e remoção de elementos nas tabelas do banco de dados.

6.4.4 Convenções e Guias para Codificação

A convenção utilizada para codificação do sistema de *software* *EduQ Student Life* é estabelecida pela Oracle no documento *Java Code Conventions*. Trata-se de boas práticas de codificação de *software* da plataforma Java, utilizada para facilitar o entendimento do código fonte e aumentar a legibilidade do *software*. A seguir algumas convenções adotadas:

Classes e interfaces: devem ser nomeadas por meio do método *Camel Case*, isto é, os substantivos devem estar juntos e a primeira letra de cada palavra deve ser maiúscula. Exemplo: “EscolaService”;

Pacotes: devem ser estruturados de acordo com a hierarquia em que se encontram. Além disto, devem ser nomeados com letras minúsculas e separados por ponto. Exemplo: “com.eduq.model.service” ;

Atributos e variáveis: Para nomeá-los, é necessário que a primeira palavra comece com letra minúscula e as demais com letras maiúsculas. Exemplo: “private List<AlunoChamada> listaChamadas” ;

Métodos: devem ser nomeados por verbos, com a primeira letra da palavra inicial com letra minúscula e as demais com letras maiúsculas. Exemplo: “public List<Classe> findAll()” ;

Identação: o alinhamento do código deve ser feito com quatro espaços do teclado (espaço correspondente à tecla *tab*).

6.4.5 Análise de Complexidade Algorítmica

A fim de verificar a eficiência de um algoritmo, ou até mesmo verificar se o mesmo se encontra correto faz-se a análise de sua complexidade. Pode-se considerar que a complexidade de um algoritmo é o esforço necessário para realizar uma tarefa, esforço este calculado com medições de funções fundamentais que o algoritmo é capaz de realizar, como por exemplo, inserções e remoções de dados.

Esta complexidade pode ser vista a princípio de duas formas diferentes, sendo elas a complexidade de código e a da solução. A análise de complexidade aplicada na plataforma *EduQ Student Life* foi justamente a de código.

A análise realizada em aspectos de eficiência leva em consideração sua complexidade temporal, o que é justamente o tempo de execução de um algoritmo em função do tamanho do problema.

No trecho de código da Figura 10 consta um método que gera uma sequência aleatória de caracteres a fim de que esses caracteres constituam uma senha provisória caso um usuário da plataforma necessite restaurar sua senha. Visto que esse algoritmo é executado apenas

uma única vez, entende-se que o mesmo possui uma classe de complexidade constante $O(1)$.

```
private String generateSecurityKey(){
    int maxCharacters = 8;
    String[] characters = { "a", "1", "b", "2", "4", "5", "6", "7", "8",
        "9", "a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k",
        "l", "m", "n", "o", "p", "q", "r", "s", "t", "u", "v", "w",
        "x", "y", "z", "A", "B", "C", "D", "E", "F", "G", "H", "I",
        "J", "K", "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U",
        "V", "W", "X", "Y", "Z", "!", "#", "$", "%" };

    StringBuilder securityKey = new StringBuilder();

    for (int i = 0; i < maxCharacters; i++) {
        int posicao = (int) (Math.random() * characters.length);
        securityKey.append(characters[posicao]);
    }

    return securityKey.toString();
}
```

FIGURA 10 - Trecho código gerador de sequência aleatórias de caracteres.

Também é possível encontrar um algoritmo com a classe de complexidade $O(N)$, ou seja, classe linear, onde o mesmo trabalha de acordo com um contador N de elementos. Na Figura 11, pode-se ver um método que trabalha nesta classe. O respectivo método realiza atualizações no banco de dados por meio de chamadas na camada DAO de acordo com o número de elementos recebidos na lista como parâmetro.

```
public void updateComponenteAlunoBoletins(List<BoletimAlunoComponente> boletins) {
    BoletimAlunoComponente aux = null;

    for(int i = 0; i < boletins.size(); i++) {
        aux = new BoletimAlunoComponente();
        aux = boletins.get(i);

        dao.updateComponenteAlunoBoletins(aux);
    }
}
```

FIGURA 11 - Método updateComponenteAlunoBoletins.

Tratando-se da análise temporal de execução de um algoritmo, foi considerado o seguinte método:

O método da Figura 12 é responsável por buscar do banco de dados um conjunto de avisos enviados pela escola, onde o retorno se constitui de uma lista de objetos da classe nativa da linguagem Java chamada *Object*, e, por sua vez, realiza uma tradução do retorno para objetos da classe Aviso.

A análise de complexidade deste método resulta na expressão $14N+3$, onde foram desconsiderados os custos de operação para as declarações de variáveis. Desta forma, os custos foram calculados considerando quatro operações na declaração da estrutura de repetição *for*, dez operações a cada passagem da repetição apenas desconsiderando a declaração do objeto aviso na primeira linha do *for*. Para completar a expressão, foram considerados mais três operações, sendo elas: a atribuição para a variável escola dos dados da atual sessão armazenada, atribuição de valores para a lista “listAux” a partir da consulta ao banco de dados, e por fim, o retorno do método.

```
@RequestMapping(value = "/getAvisos", method = RequestMethod.GET)
public List<Aviso> getAvisos(HttpSession session) {
    Escola escola = new Escola();
    List<Object[]> listAux = new ArrayList<>();
    List<Aviso> avisos = new ArrayList<>();
    Aviso aviso = null;
    Object[] result = null;

    escola = (Escola) session.getAttribute("escolaLogada");
    listAux = avisoService.buscarPorEscola(escola.getId());

    for(int i = 0; i < listAux.size(); i++) {
        aviso = new Aviso();

        result = listAux.get(i);

        String dataArray[] = result[3].toString().split("-");
        String data = "" + dataArray[2] + "/" + dataArray[1] + "/" + dataArray[0] + "";

        aviso.setId(Long.parseLong(result[0].toString()));
        aviso.setAnoClasse(result[1].toString());
        aviso.setConteudo(result[2].toString());
        aviso.setDataAviso(data);
        aviso.setTipoAviso(result[5].toString());
        aviso.setTituloAviso(result[6].toString());

        avisos.add(aviso);
    }

    return avisos;
}
```

FIGURA 12 - Método busca dados conjunto de aviso enviado pela escola.

6.5 PROJETO DE SISTEMA DISTRIBUÍDO

Um sistema distribuído é um conjunto de computadores independentes que se apresenta a seus usuários como um sistema único e coerente. Não há premissa ou uma configuração necessária com relação aos computadores do sistema distribuído, de forma que estas configurações, bem com sua distribuição de processamento, devem ser transparentes aos usuários, que devem ter a percepção de estar utilizando um único sistema. (TANENBAUM, 1999).

6.5.1 Procedimentos para Tratamento dos Desafios

Nesta seção, apresentam-se obstáculos que, eventualmente, podem vir a acontecer no desenvolvimento e implementação de um sistema distribuído, de forma que os mesmos sejam superados a fim de definir uma qualidade geral ao sistema.

6.5.1.1 Heterogeneidade

O sistema distribuído é composto de diversos *hosts* interconectados, onde cada um pode possuir um sistema operacional diferente, que por sua vez pode executar um cliente que foi escrito em uma linguagem de programação diferente da qual foi utilizada no desenvolvimento do serviço no processo servidor. Justamente pela existência de diferentes nós computacionais dentro do mesmo sistema distribuído, este se torna um obstáculo para o mesmo, sendo necessário neste projeto, o desenvolvimento de uma API(*Application Programming Interface*) padronizada para consultas e operações de escrita dentro da arquitetura.

6.5.1.2 Escalabilidade

A escalabilidade vem como um componente importante da implementação de um sistema distribuído, o qual estando em rede, pode ter um aumento ou diminuição de possíveis *hosts* clientes dentro da arquitetura. Esta possibilidade pode acarretar perda de poder de processamento do sistema. Desta forma, a arquitetura do sistema distribuído deve ser dinâmica a ponto de suportar a conexão e desconexão de clientes. Para o EduQ *Student Life*, o plano para trabalhar com este desafio se baseia na implementação de algoritmos otimizados, bem como na contratação de servidores com serviços de alocação de recursos dinâmicos no momento da implantação.

6.5.1.3 Abertura

A abertura no âmbito de um sistema distribuído se faz importante na manutenção ou realização de novas implementações. Para isso, os métodos devem ser padronizados e novos métodos adicionados devem ter a capacidade de se comunicarem com os já existentes sem causar-lhes danos. Para vencer este obstáculo, o EduQ *Student Life* deve possuir uma

documentação publicada sobre seus métodos públicos, facilitando possíveis novas implementações.

6.5.1.4 Segurança

Os aspectos de segurança devem ser considerados na problemática quando se projeta um sistema distribuído. Requisitos como confidencialidade, autenticidade, integridade e disponibilidade devem ser tratados de forma especial pelo sistema. Quanto ao obstáculo da confidencialidade, o *EduQ Student Life* pode fazer uso de métodos criptográficos em suas transações. Na autenticidade, o *EduQ Student Life* deve utilizar certificados digitais tanto no cliente *Web*, quanto na aplicação *mobile*. Em consideração à integridade, os dados devem ser checados antes de qualquer transação ser realizada, evitando dados fantasmas e perda de conteúdo dentro do sistema. Quanto à disponibilidade, o *host* servidor deve possuir um controle de *proxy*, o qual deve estar constantemente atualizado bloqueando assim ataques, como o de negação de serviço.

6.5.1.5 Manuseio de Falhas

O controle das falhas dentro de um sistema distribuído é certamente um dos maiores desafios em seu desenvolvimento. Uma vez que existem muitos componentes computacionais dentro da mesma arquitetura, a observação e o tratamento quanto às falhas devem ser constantes e devem ser informados aos administradores, para que os mesmos possam vir a tomar decisões e entrarem com medidas corretivas. Dentro do *EduQ*, podem ser implementados mecanismos de *log* com falhas de processamento, de forma que este arquivo possa ser consultado para aplicar ações por parte dos administradores.

6.5.1.6 Concorrência

Com o aumento do número de clientes dentro do sistema distribuído, é possível que dois venham a tentar acessar o mesmo recurso, e a arquitetura deve ser projetada de forma que todas as requisições sejam respondidas, positiva ou negativamente. O *EduQ Student Life* deve possuir um controle de requisições efetuadas, principalmente em relação a seu banco de dados, o qual deve permitir que apenas um usuário manipule o recurso por vez, a fim de evitar consultas e transações inconsistentes no banco de dados e nas aplicações.

6.5.1.7 Transparência

Como em um sistema distribuído, os recursos computacionais estão localizados em diferentes componentes, o acesso aos mesmos deve ser facilmente realizado, sendo transparente ao cliente o fato de o mesmo estar acessando recursos locais ou remotos. O *EduQ Student Life* deve permitir que o cliente acesse os recursos de forma fácil e ágil, não fazendo distinção sobre a localidade do componente no qual o recurso está localizado.

6.5.2 Tecnologias e Arquiteturas de Distribuição

Para a programação dos componentes do sistema de *software* EduQ, a tecnologia *Java Enterprise Edition* (JEE) é adotada. Esta é voltada para redes, Internet e afins e contém bibliotecas e funções específicas para a portabilidade que o mercado tem consumido atualmente.

Com esta tecnologia, é possível estabelecer uma arquitetura cliente-servidor. Neste modelo, os processos do sistema distribuído são divididos em dois grupos: clientes e servidores. Um servidor é um processo que implementa um serviço específico, já um cliente, é um processo que requisita um serviço de um servidor enviando-lhe uma requisição e aguardando uma resposta do servidor (TANENBAUM, 1999).

Baseado nesta arquitetura, na Figura 13, os *hosts* 1 e 3 (H1 e H3) executam processos clientes e os *hosts* 3 e 4 (H3 e H4) executam os diversos processos servidores.

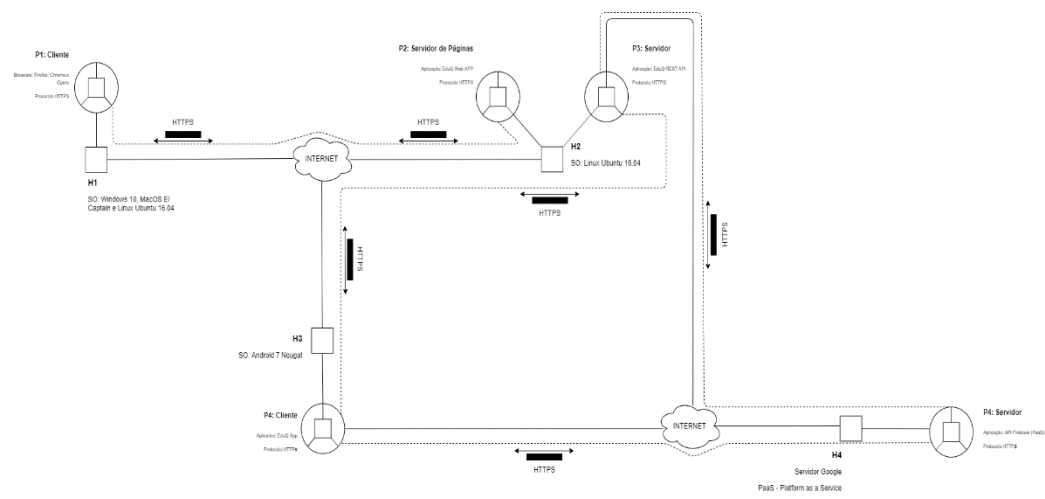


FIGURA 13 - Diagrama do sistema distribuído.

6.5.2.1 Google Firebase

O Google Firebase é um *framework* para desenvolvimento *web* e *mobile* que contém diversas ferramentas para construção de sistemas de software (GOOGLE, 2019). As utilizadas neste projeto são:

- a. *Cloud Firestore*: um banco de dados de documentos NoSQL (*Not Only Structured Query Language*) que permite armazenar, sincronizar e consultar dados persistidos em nuvem;
- b. *Authentication*: visa facilitar o desenvolvimento de um sistema de autenticação seguro e melhorar a experiência de acesso e ambientação para os usuários finais. Ele oferece uma solução de identidade completa, compatível com contas de e-mail/senha, autenticação por telefone, autenticação do Google, Twitter, Facebook, GitHub e outros;
- c. *Cloud Messaging*: oferece uma conexão confiável e com baixo consumo de bateria entre servidor e dispositivos para enviar e receber mensagens e notificações no Android, no iOS e na *web*.

6.6 PROJETO DA INTERAÇÃO HUMANO-COMPUTADOR

A interação humana com o sistema de *software* acontece por meio da interface de usuário e o sistema deve se ser capaz de se adaptar aos diversos perfis de usuários, desde os novatos até os experientes.

As áreas de uma interface precisam ser bem definidas para proporcionar uma boa interação humana-computador. As dúvidas que surgem na navegação do usuário distraem a atenção da tarefa executada, que se acumuladas, poderão confundi-lo. Toda a solução visual deve ser bem clara para a percepção do usuário, que permite acelerar o processo de navegação, tornando o sistema bem mais funcional e intuitivo.

6.6.1 Perfil de Usuário

A definição dos perfis de usuários esperados para a utilização do sistema EduQ *Student Life* permite a inclusão, remoção e atualização de seus acessos, bem como construir interfaces mais apropriadas para cada um deles, a saber:

- a. Escola: colaboradores que irão realizar o cadastro dos professores, alunos, classes, componentes curriculares e também a atualização dos elementos relacionados;
- b. Professor: realizará chamadas diárias, enviará avisos e irá atribuir conceitos a cada aluno da classe;
- c. Pais ou responsável: farão uso do sistema por meio de um aplicativo móvel, onde poderão acompanhar a vida escolar da criança e também enviar avisos e requisições para a escola e ao professor.

As entrevistas com os futuros e prováveis usuários encontram-se no Apêndice P.

6.6.2 Aspecto Visual da Interface de Usuário

Em um mundo altamente tecnológico, a usabilidade deve ser muito bem apresentada. Segundo Pressman (2011, p. 287), a usabilidade é uma medida qualitativa sobre a facilidade e eficiência com que o ser humano opera as funções e recursos oferecidos por um produto tecnológico. Pressman (2011, p. 287) ainda reforça sua importância dizendo que caso um sistema de *software* seja difícil de ser utilizado, ele irá frustrar o usuário em seus esforços para atingir suas metas, independente do poder computacional e conteúdos fornecidos. Assim, a interface molda a percepção do *software* pelo usuário.

A fim de alcançar uma boa percepção de seus usuários com relação aos componentes computacionais do sistema, o EduQ *Student Life* apresenta um projeto de HCI com aspectos minimalistas e uma paleta de cores suaves.

Uma interface minimalista é aquela que traz ao usuário apenas informações que são relevantes para seu uso, evitando excessos de conteúdo irrelevantes ou que são raramente utilizados.

Ainda se tratando de interfaces, o EduQ *Student Life* opta por também trazer uma abordagem simples intuitiva quanto aos ícones utilizados. Vale também ressaltar que os

ícones são partes importantes para conferir usabilidade, pois eles não só auxiliam no entendimento do contexto, como também indicam possíveis rotas para a tomada de decisões dentro da aplicação.

A cor é considerada o elemento visual da interface que influencia diretamente na qualidade da apresentação das informações transmitidas, desta forma, evidencia-se sua contribuição na usabilidade de uma interface computacional de usuário (KULPA; PINHEIRO; SILVA, 2011, p.124).

Segundo Nielsen e Loranger (2007), o vermelho mostra-se muito eficiente quando usado nas interfaces para chamar a atenção ou sinalizar algum perigo, porém seu uso deve ser evitado em áreas amplas ou como cor de fundo, pois se trata de uma cor dominante. Os autores também afirmam que o branco é usado pelo olho para determinar o conteúdo espectral de um iluminante, indicando assim, o fundo branco para uma interface, pois fornece a máxima legibilidade para um texto escuro e, portanto, é a cor mais amplamente utilizada para o fundo dos sites, apesar de seu intenso brilho causar problemas ao usuário que permaneça muito tempo em contato visual com a mesma (KULPA; PINHEIRO; SILVA, 2011, p.128).

Com base nos conceitos e detalhes citados, o EduQ *Student Life* traz a paleta de cores representada na Figura 14, cores estas que são utilizadas tanto no componente *Web*, quanto no componente *mobile*.



FIGURA 14 - Paleta de cores utilizada no EduqWeb e EduqApp.

A Figura 15 mostra um menu lateral do EduqWeb, o qual possui em suas cores o azul claro e o branco, bem como os ícones simples e intuitivos.

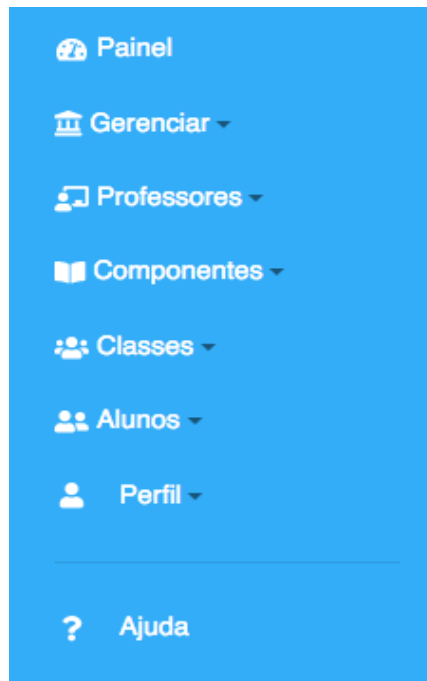


FIGURA 15 - Menu lateral do EduqWeb.

6.6.3 Heurísticas de Usabilidade

Heurística é a arte de descobrir e inventar, que por sua vez se torna uma técnica na qual se pode utilizá-la como atalho mental a respeito de determinados assuntos. Nielsen (1994) propôs algumas heurísticas sobre o desenvolvimento de interfaces de usuário, as quais devem ser consideradas no projeto da HCI. Essas heurísticas são apresentadas nas próximas subseções.

6.6.3.1 Visibilidade de qual estado o sistema se encontra

Um usuário deve saber em qual parte do sistema está e sobre o que sistema está tratando no momento. Na Figura 16, o EduQWeb apresenta, no canto superior esquerdo, a localização do usuário, bem como ações que ele poderá ativar no momento.

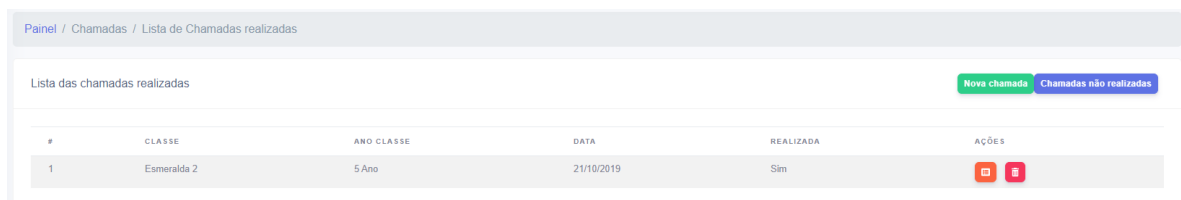
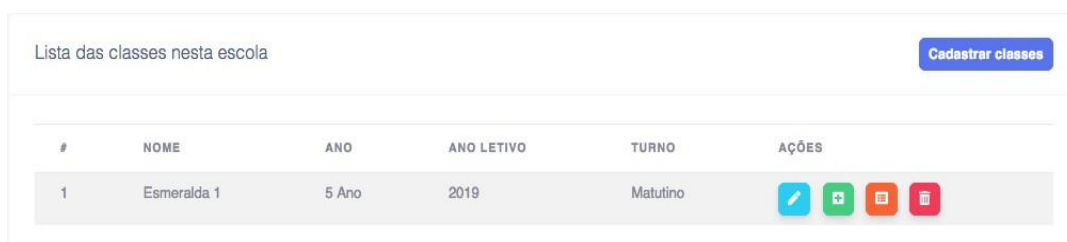


FIGURA 16 - Lista de chamada no EduqWeb.

6.6.3.2 Correspondência entre o sistema e o mundo real

A fim de facilitar o aprendizado da aplicação pelo usuário, a interface de usuário deve ser intuitiva e apresentar elementos do mundo real, tanto em seu texto verbal quanto por meio de itens gráficos (como os ícones). A Figura 17 mostra a exemplificação de como o EduQWeb expõe itens que refletem o mundo real, facilitando o aprendizado por parte do usuário. O ícone de “Editar” é evidenciado por um lápis, a ação “Adicionar Alunos” por um sinal de adição, a ação “Lista de Alunos” por meio de uma lista e ação “Excluir” é acionada por um ícone com aparência de uma lixeira.



#	NOME	ANO	ANO LETIVO	TURNO	AÇÕES
1	Esmeralda 1	5 Ano	2019	Matutino	[Ícone de lápis] [Ícone de +] [Ícone de lista] [Ícone de lixeira]

FIGURA 17 - Lista de classes abertas pela escola no EduQWeb.

6.6.3.3 Liberdade e controle para o usuário

A finalidade desta heurística é justamente dar ao usuário o poder de trabalhar na aplicação da forma que entender, cabendo a ele decidir se quer mesmo ou não realizar determinada ação. Na Figura 18, o EduQWeb aplica esta heurística. Uma vez que o usuário opte por excluir determinado item dentro da aplicação, ele irá se deparar com uma mensagem pedindo para que ele confirme se deseja completar a ação, com a opção de cancelamento, caso o usuário desista dessa operação.

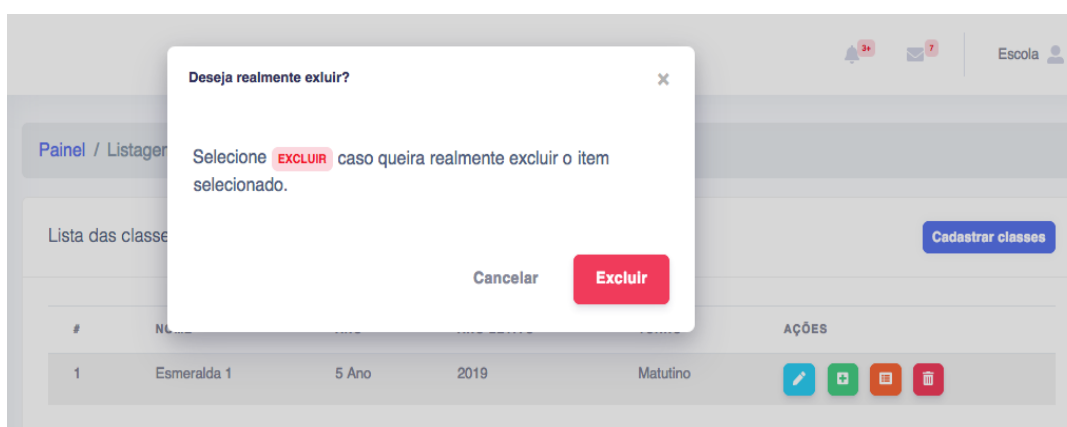
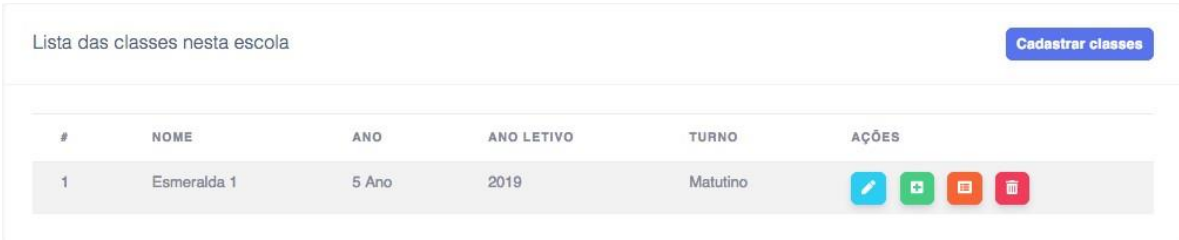


FIGURA 18 - Mensagem de confirmação para a exclusão.

6.6.3.4 Consistência e padrões

Esta heurística diz respeito à importância de manter padronizados os aspectos visuais do sistema, sejam eles, cores, posicionamento de elementos ou exibição de conteúdo na tela. A consistência faz com que se amplie a familiaridade do usuário com a interface e que ele se sinta mais confortável em utilizar a aplicação. Nas Figuras 19 e 20, há dois exemplos que exemplificam duas listas diferentes dentro do EduQWeb, para as quais se utilizam os mesmos padrões para exibir as informações.







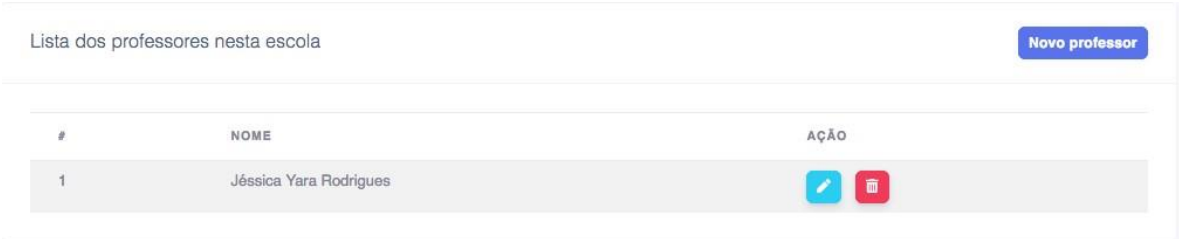
#	NOME	ANO	ANO LETIVO	TURNO	AÇÕES
1	Esmeralda 1	5 Ano	2019	Matutino	   

FIGURA 19 - Lista de classes cadastradas no EduqWeb.





#	NOME	AÇÃO
1	Jéssica Yara Rodrigues	 

FIGURA 20 - Lista dos professores cadastrados no EduqWeb.

6.6.3.5 Prevenções de erros

Uma interface bem definida e projetada não deve deixar que o usuário cometa ações que possam afetar sua eficiência de uso ou provocar erros de sua parte. Sendo assim, todas as ações da aplicação devem ser avisadas ao usuário sobre sua criticidade, informando-o que a ação pode não ser desfeita. A Figura 21 mostra como o EduQWeb apresenta alertas ao usuário.

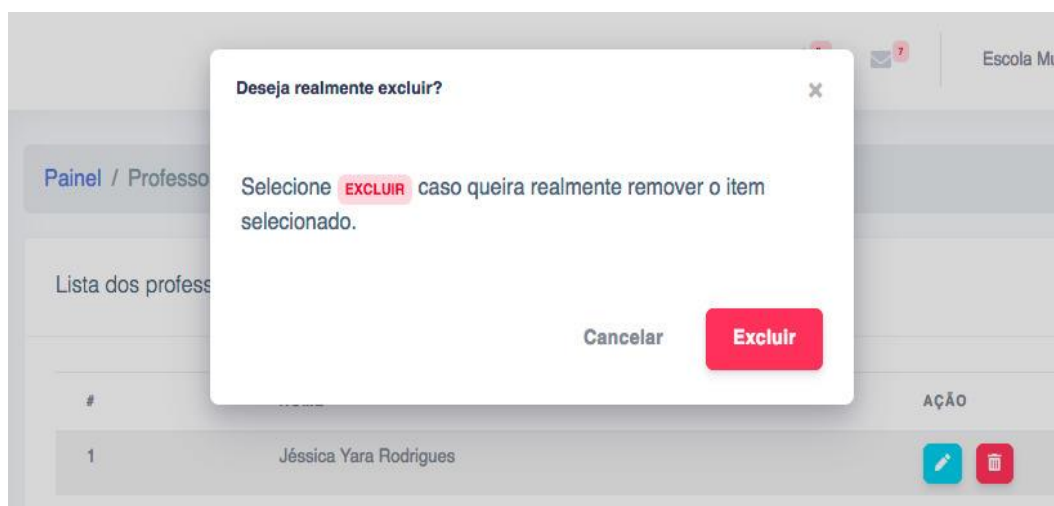


FIGURA 21 - Mensagem de alerta ao usuário.

6.6.3.6 Reconhecimento em vez de memorização

Uma boa interface de usuário não deve forçar a lembrança de todo o caminho necessário para ativar uma ação. Ela deve fornecer algum tipo de sinalização para esse caminho. A Figura 22 mostra a técnica de *breadcrumbs* (sequência do caminho percorrido), que permite ao usuário identificar em que parte do sistema ele está.

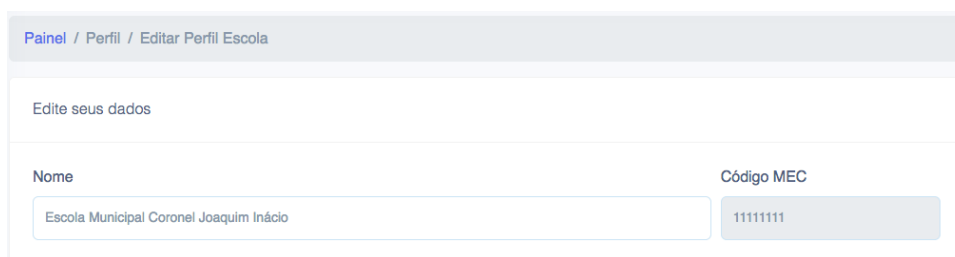


FIGURA 22 - Técnica do *breadcrumb* aplicada pelo EduqWeb.

6.6.3.7 Flexibilidade e eficiência de uso

É necessário que o usuário possua mecanismos que o ajude em sua utilização e agilize o seu manuseio da aplicação. As Figuras 23 e 24 mostram a maneira pela qual o EduQWeb trabalha com este aspecto, oferecendo um menu lateral que estará sempre visível no uso da aplicação e também atalhos disponíveis na página principal.

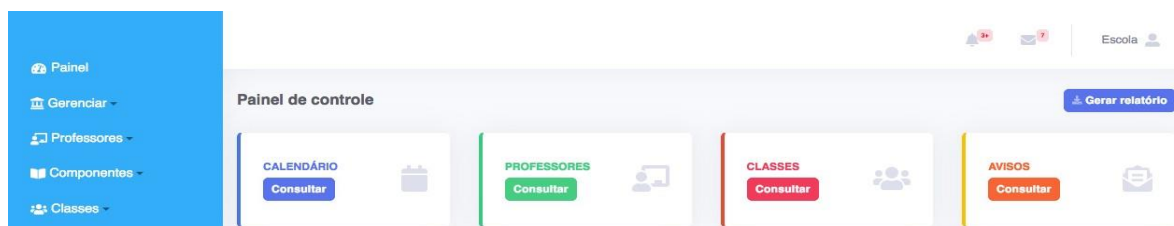


FIGURA 23 - Menu e atalhos.

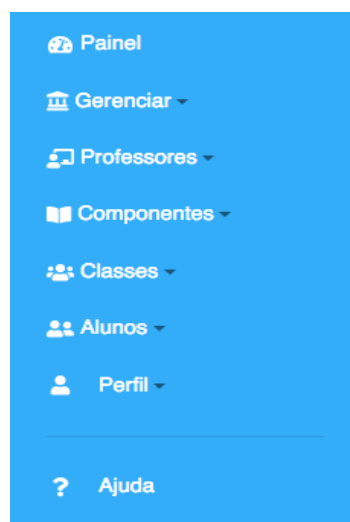


FIGURA 24 - Menu lateral do EduqWeb.

6.6.3.8 Estética e design minimalista

Todo tipo de informação exibida nas interfaces deve ser projetado de forma que não interfira no uso da aplicação pelo usuário. Essas informações devem estar em evidência na página de modo que o foco de atenção do usuário fique explicitamente nelas. Distrações fazem com que o usuário perca o foco, o que consequentemente poderá induzi-lo às falhas. A Figura 25 exibe a página de autenticação do EduQWeb e mostra para o usuário os dados que ele necessita preencher para que tenha o acesso liberado à aplicação.

 A imagem mostra a página de autenticação do EduQWeb. O formulário tem um cabeçalho cinza com o texto 'Autenticação EduQ'. Abaixo dele, há dois campos de entrada: 'Código MEC' e 'Senha'. Um botão azul com o texto 'Entrar' está abaixo dos campos. Abaixo do botão, há uma caixa de seleção com o texto 'Lembrar?'. Na base do formulário, há três links azuis: 'Página principal', 'Esqueceu sua senha?' e 'Cadastre-se'.

FIGURA 25 - Página de autenticação do EduqWeb.

6.6.3.9 Reconhecimento e recuperação de erros

O tratamento dos erros cometidos pelo usuário deve ser bem definido de forma que haja dentro da aplicação uma informação ao usuário sobre o que aconteceu na operação que causou o problema. Neste cenário, o EduQWeb, em sua opção de alteração de senha, solicita

ao usuário que informe sua senha atual, a nova senha com a sua respectiva confirmação. Caso o usuário informe uma senha divergente da atual, é apresentada uma mensagem de erro explicativa, na qual é evidenciada que a senha fornecida não confere com a realidade. A Figura 26 mostra como essa operação é feita.

Painel / Perfil / Atualizar senha

Necessário informar a senha atual. Tente novamente.

Preencha os dados abaixo para alterar sua senha

Informe sua senha atual

Informe a nova senha

Confirme sua senha

Senha atual

Nova senha

Repita a senha

Salvar

FIGURA 26 - Página para alterar a senha de acesso ao EduqWeb.

6.6.3.10 Ajuda e documentação

Dúvidas podem surgir por parte do usuário quanto à utilização da aplicação, principalmente, no início. Com isso, os sistemas devem se atentar em trazer ferramentas de auxílio para os usuários dentro do escopo de uso. Nas Figuras 27 e 28, nota-se que o EduQWeb traz uma seção de ajuda no menu lateral, o que possibilita ao usuário ler instruções de operação sobre as partes críticas do sistema, bem como a existência de um formulário no qual ele poderá utilizá-lo para enviar sua dúvida diretamente aos administradores do sistema.

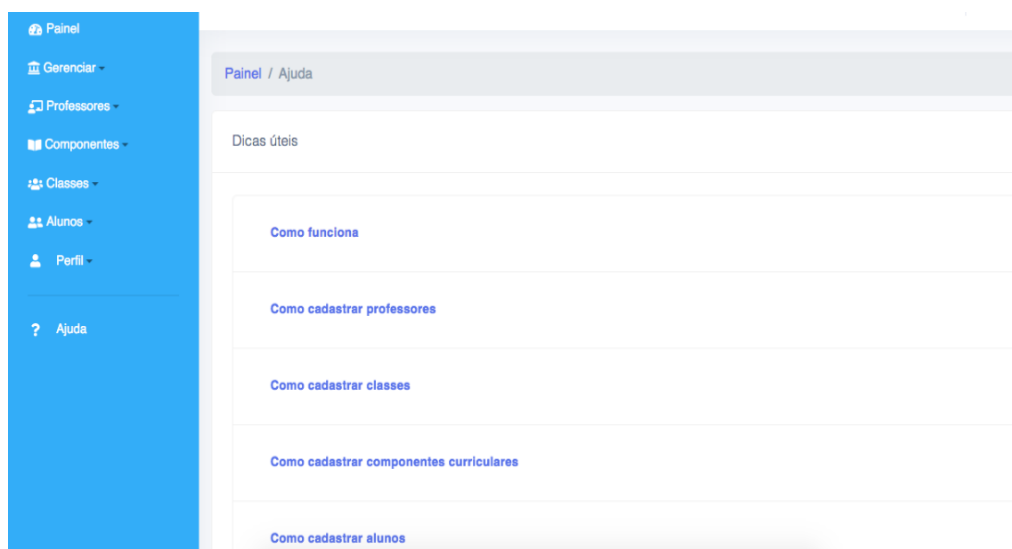


FIGURA 27 - Página de ajuda do EduqWeb.

Não encontrou o que procurava?

Fique tranquilo, preencha o formulário abaixo com a sua dúvida.

Este formulário será enviado diretamente para nós, que vamos responder o mais rápido possível no e-mail que você usou para se cadastrar.

Título

Escreva um título para sua dúvida

Descrição

Descreva sua dúvida aqui

Enviar

FIGURA 28 - Formulário para preenchimento de dúvidas no EduqWeb.

6.6.3.11 Exemplos de interface de usuário do EduQApp

Na Figura 29, é mostrada a tela de autenticação do aplicativo móvel EduqApp. Por ela, o responsável poderá escolher se autenticar utilizando conta do sistema ou conta do Google.



FIGURA 29 - Tela de autenticação do EduqApp.

6.6.3.12 Perfil do Usuário EduqApp

A Figura 30 mostra a tela de perfil do usuário onde devem ser adicionados os alunos que têm códigos gerados pela escola. Os dados pessoais podem ser editados e a lista dos alunos que está sob a responsabilidade do usuário.



FIGURA 30 - Tela do perfil de usuário do EduqApp.

7 PLANO DE TESTES

Neste capítulo são mostradas algumas das estratégias de testes adotadas neste projeto, qual sua finalidade, quais artefatos são testados, como os testes são executados e quais os resultados obtidos.

Segundo Bartié (2002), o plano de testes formaliza a estratégia de testes, contendo informações sobre o escopo de testes, configuração de ambiente, recursos necessários e cronograma de testes.

Os erros e falhas no sistema de *software* podem influenciar, de forma negativa, a percepção do usuário final sobre o produto utilizado. Assim, a fim de evitar que erros e falhas ocorram, ou fazer com que ocorram ainda em ambiente de desenvolvimento, não sendo percebidos pelos usuários, a equipe de desenvolvimento deve adotar estratégias de testes.

7.1 FINALIDADE

O plano de testes aqui apresentado tem a finalidade testar algumas das funcionalidades do EduQWeb.

7.2 ESCOPO

Esta seção apresenta todos os documentos e *softwares* necessários para a realização dos testes planejados.

7.2.1 Referências a Documentos Relevantes

O Quadro 6 mostra o documento necessário para a elaboração do plano de testes.

Tipo do Material	Referência
EduQ <i>Student Life</i> .docx	Capítulo 5 - Especificação e Análise de Requisitos.

QUADRO 6 - Documento relevante para testes

7.2.2 Ambiente para a Realização dos Testes

O Quadro 7 mostra o equipamento necessário para a realização dos testes planejados.

Equipamento	Marca/Modelo/Configuração	Finalidade
Notebook	Processador Dell Core i7-8550U; memória RAM de 8 GB; Sistema Operacional Windows 10 Pro 64 bits.	Executar os testes de integração e os testes caixa preta no componente Web.

QUADRO 7 - Equipamentos para a realização dos testes

O Quadro 8 mostra os *softwares* necessários para a realização dos testes planejados.

Software/Versão	Fabricante	Finalidade
Eclipse IDE	IBM / 2018-9 4.9.0	Execução do sistema para realização dos casos de teste.
PostgreSQL	PostgreSQL Global Development Group /4.6.0.0	Sistema gerenciador de banco de dados que fornece a base de dados para os componentes.
Selenium Web Driver	Selenium	Gerador de testes automatizados.
Google Chrome	Google LLC/76.0.3809.100	Aplicação cliente que será utilizada para entrada e saída de dados.

QUADRO 8 - Softwares para a realização dos testes

7.3 ESPECIFICAÇÃO DOS CASOS DE TESTE

Esta seção define o conjunto de casos de teste a serem realizados, os itens a serem testados, especificação das entradas e saídas esperadas, rastreabilidade entre requisitos e as descrições dos casos de testes.

7.3.1 Item a Testar

O Quadro 9 mostra os itens a serem testados e uma descrição de cada item.

Identificação do Item	Descrição
1	RF01 – Cadastrar Escola
2	RF02 – Efetuar Autenticação da Escola
3	RF03 – Cadastrar Professor
4	RF05 – Efetuar Autenticação do Professor
5	RF06 – Alterar Senha
6	RF07 – Redefinir Senha
7	RF10 – Efetuar Saída
8	RF13 – Cadastrar Aluno
9	RF14 – Excluir Aluno
10	RF15 – Alterar Cadastro de Aluno

QUADRO 9 - Identificação dos itens a serem testados

7.3.2 Rastreabilidade entre Requisitos e Casos de Teste

Para cada item a ser testado, deve-se elaborar um conjunto de casos de teste que abordam todas, ou quase todas, as possibilidades dentro de um cenário. A rastreabilidade, ou seja, a relação de cada item com seus casos de teste está disponível no Apêndice W.

7.3.3 Descrição dos Casos de Teste

Um item a ser testado pode conter um ou mais casos de teste. Sendo que cada caso de teste aborda um possível cenário diferente de execução. Segundo Endler (2018), um caso de teste:

mostra os caminhos percorridos por um módulo, caso de uso ou funcionalidade dentro do projeto. Serve como base para que os testadores possam executar os testes manualmente, mas pode ser criado, também, com o intuito de automatizar os testes. Um item a ser testado pode conter um ou mais casos de teste. Sendo que cada caso de teste aborda um possível cenário diferente de execução.

As ações são os passos que serão executados pelo testador. Estas ações devem levar a um resultado esperado. A Figura 31 ilustra os diversos fluxos de eventos que um caso de uso pode ter.



FIGURA 31 - Fluxos básicos e alternativos.

Fonte: Endler (2018).

Assim sendo, o Apêndice X contém a descrição dos casos de teste elaborados para os itens citados na Seção 7.3.1.

7.4 RESULTADOS DOS TESTES

Nesta seção são apresentados os resultados obtidos com a execução dos casos de teste previstos neste capítulo.

7.4.1 Histórico de Realização

No histórico de produção dos casos de teste encontra-se o nome do responsável pelo teste, a data de prática do teste e o resultado obtido. O Apêndice Y contém o histórico de realização de uma bateria dos testes aqui previstos.

7.4.2 Resultados Obtidos

O plano de testes elaborado para este projeto inclui algumas funcionalidades principais do componente *EduqWeb*. Com os testes descritos neste capítulo é possível obter um resultado satisfatório para a conclusão do projeto e ter a garantia de que as funcionalidades atendem ao que foi especificado como requisito.

8 PLANO DA IMPLANTAÇÃO

Este capítulo apresenta a finalidade do plano de implantação, a metodologia adotada para a implantação, a matriz de responsabilidades, os treinamentos previstos, o cronograma de implantação, os documentos de apoio a implementação e a visão da implementação.

8.1 METODOLOGIA

Nesta seção são apresentadas a descrição da metodologia de implantação utilizada e a matriz de responsabilidades.

8.1.1 Descrição da Metodologia

Para o funcionamento do sistema de forma eficaz, o *Eduq Student Life* deve ser implantado seguindo os seguintes passos:

EduqWeb:

- a) designar um responsável pela implantação;
- b) realizar a configuração do servidor de *Virtual Private Server*;
- c) criar uma instância do banco de dados no servidor PostgreSQL;
- d) executar os *scripts* de criação do banco de dados;
- e) adicionar a componente web *service* no servidor Tomcat;
- f) adicionar o componente EduqWeb no servidor.

EduqApp:

- a) disponibilizar o arquivo apk na Google Play Store, como o nome de: *EduQ Student Life*.

8.1.2 Matriz de Responsabilidade

O Quadro 10 apresenta as atividades preparatórias para a implantação e as responsabilidades do cliente e da equipe de implantação do sistema de *software*.

Atividades	Responsáveis
Planejamento	
Definição da equipe de implantação	Equipe
Levantamento de recursos necessários de <i>hardware</i>	Felipe
Levantamento de recursos necessários de <i>software</i>	Gabriel
Definição de estratégias para conversão e migração de dados	Gabriel
Programação dos treinamentos	Christian
Preparação dos testes de aceitação	Christian
Execução	
Configuração da infraestrutura de Tecnologia da Informação	Gabriel
Instalação	Felipe
Conversão e migração de base de dados	Felipe
Treinamento	Christian
Realização de testes de aceitação	Christian
Avaliação	
Acompanhamento pós-implantação	Equipe
Reunião final de implantação	Equipe

QUADRO 10 - Papéis e responsabilidades na implantação

8.2 TREINAMENTO PREVISTO

O Quadro 11 detalha os treinamentos a serem ministrados para a capacitação dos usuários durante a fase de implantação do sistema.

Treinamento	Conteúdo	Grupo de Usuários
Treinamento prático de utilização do EduqWeb para administração da instituição de ensino.	Operação com a interface de usuário do sistema de <i>software</i> e regras básicas para operação do sistema na instituição, cadastro de professores e alunos	Administradores da instituição.
Treinamento prático de utilização do EduQWeb para professores da instituição de ensino.	Operação com a interface de usuário do sistema de <i>software</i> e regras básicas para operação do sistema na instituição, aprendizado do método enviar avisos aos pais, controle de frequência e boletim de notas e conceitos.	Professores da instituição de ensino.
Treinamento via manual para responsáveis pelos alunos utilizarem o EduQApp.	Operação com interface de usuário do aplicativo móvel e regras básicas para operação e consulta a avisos de professor, verificar faltas, notas, conceitos e trabalhos.	Responsáveis pelo aluno.

QUADRO 11 - Treinamentos previstos

8.3CRONOGRAMA DE IMPLANTAÇÃO

O tempo de realização dos treinamentos e instalação do sistema deverão ser definidos pelo responsável da implantação juntamente com o cliente (instituição de ensino).

O Quadro 12 são apresentadas as tarefas previstas durante a implantação, a duração em horas e o período de realização.

Tarefas	Duração	Período
Instalação do servidor.	24 horas	A definir
Configuração do banco de dados.	4 horas	A definir
Configurações necessárias de integração.	4 horas	A definir
Treinamento dos usuários.	10 horas	A definir
Tempo estimado total:	42 horas	

QUADRO 12 - Cronograma de atividades da implantação

8.4 DOCUMENTOS DE APOIO À IMPLANTAÇÃO

Os documentos oferecidos para apoiar o processo de implantação e posterior uso do sistema são listados no Quadro 13.

Documento	Referência
Manual do usuário do EduqWeb para instituição de ensino e professores	Vide Apêndice Z.
Manual do usuário do EduqApp para os usuários responsáveis (pais ou responsável legal).	Vide Apêndice Z.

QUADRO 13 - Documentos de apoio à implantação

8.5 VISÃO DA IMPLANTAÇÃO

Para um melhor entendimento deste plano de implantação, no Apêndice AA consta o Diagrama de Implantação da UML para o sistema *EduQ Student Life*.

9 CONCLUSÃO

Na primeira fase do projeto foram definidas as funcionalidades, restrições, objetivos, público alvo e as tecnologias a ser utilizadas para o desenvolvimento do sistema. Para auxiliar nessa etapa, foi realizada uma pesquisa bibliográfica em busca de embasamento teórico e uma entrevista com educadores do Ensino Fundamental. Também nessa fase, iniciou-se a modelagem do sistema pelo modelo de entidade e relacionamento, de casos de uso e elaboração dos planos de gerência do projeto. O maior obstáculo encontrado nesta fase foi a falta de tempo e alguns contratempos em relação à saúde debilitada de alguns membros da equipe. Essa dificuldade refletiu no desempenho e na entrega da fase.

Durante a segunda fase do projeto, foram realizadas correções em busca da melhoria do projeto. Quanto à arquitetura do sistema de software, iniciou-se a construção dos artefatos que compõem as visões estrutural, comportamental e de dados. Também foi iniciada a codificação dos componentes do sistema de software *Eduq Student Life*.

Durante as duas fases iniciais do projeto, a equipe teve o apoio da professora Jéssica Yara Rodrigues que auxiliou fornecendo informações e sanando dúvidas. A entrevista feita com a diretora da Escola Municipal Cel. Joaquim Inácio, profa. Fabiane Thomaz Macedo, proporcionou um melhor entendimento e conhecimento do ambiente educacional real. Com isso, a parceria estabelecida entre os profissionais de educação e esta equipe enriqueceu os conceitos teóricos e técnicos deste projeto.

Durante a terceira fase do projeto, foram realizadas outras melhorias no projeto. A técnica de pontos de função foi utilizada, o que permitiu à equipe obter uma estimativa de esforço para o desenvolvimento do projeto. Nessa fase também foi possível obter uma versão funcional do sistema de software e aplicar testes automatizados.

Durante a quarta fase do projeto, definiu-se um plano de implantação do sistema, incluindo a produção de manuais para os usuários. O sistema *Eduq Student Life* foi apresentado ao público na primeira noite da XXX Feira de Tecnologia da FAI (Faitec). Infelizmente, a Faitec foi cancelada devido a uma forte chuva de granizo que causou danos a todo o povo de Santa Rita do Sapucaí.

Em relação aos objetivos do projeto, a Figura 32 mostra o aplicativo móvel EduqApp a ser utilizado pelo responsável do aluno para receber avisos e eventos comunicados pela instituição de ensino.

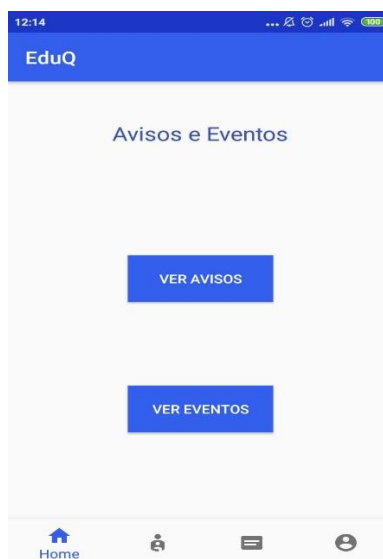


FIGURA 32 - Tela de avisos e eventos do EduqApp.

A Figura 33 mostra como os responsáveis conseguem acompanhar informações escolares, como presenças em classe, avisos, boletins e trabalhos.



FIGURA 33 - Tela de acompanhamento EduqApp.

A Figura 34 mostra o aplicativo web EduqWeb por meio do qual o profissional da Educação conseguirá enviar comunicados e divulgar eventos aos responsáveis pelo aluno.

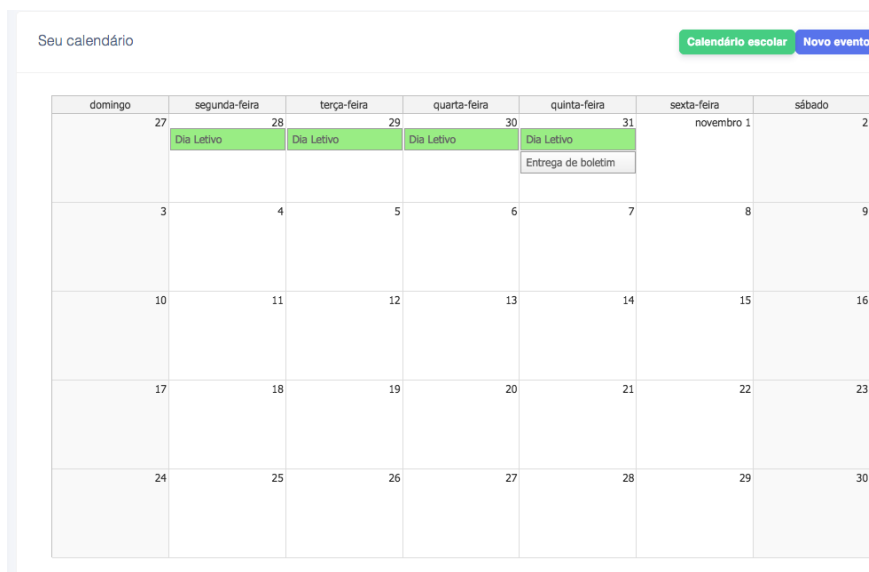


FIGURA 34 - Tela de consulta do calendário escolar.

Como proposta de evolução do sistema Eduq, espera-se criar um ambiente de testes dentro de uma instituição de Ensino Fundamental para operar o sistema e coletar *feedback* dos usuários. E, também refinar o sistema em relação aos controles mais específicos desse nível de ensino.

O desenvolvimento deste projeto permitiu aos integrantes da equipe vivenciar os desafios e dificuldades encontradas com o projeto real. Pode-se concluir que as situações vividas no decorrer deste projeto, de forma geral, foram muito positivas para o crescimento pessoal e profissional de todos os integrantes da equipe. Contudo, destaca-se a aprendizagem adquirida, especialmente, nas áreas de gestão do tempo, de recursos humanos e de comunicação.

REFERÊNCIAS

BARTIÉ, A. **Garantia da Qualidade de Software**. 5. ed. Elsevier, 2002

BRASIL. Constituição (1988). **Constituição da República Federativa do Brasil**: promulgada em 5 de outubro de 1988. Brasília: Diário Oficial da União, 1988.

BRASIL. **Base Nacional Comum Curricular**. Brasília: Ministério da Educação, 2018.

BATEMAN, T.; SNELL, S. **Administração**: construindo vantagem competitiva. São Paulo: Atlas, 1998.

BEZERRA, E. **Princípios de análise e projeto de sistema com UML**. Rio de Janeiro: Elsevier, 2002. 286p.

DEMO, P. **TICs e educação**, Belo Horizonte: [S. n.], 2017. Disponível em: <<http://www.pedrodemo.sites.uol.com.br/>>. Acesso em: 29 mai. 2019.

DOMICIANO JUNIOR, A.; OLIVEIRA, G.; CASTRO, L. T. **Emother**. 2015. Projeto Final de Curso (graduação em Sistemas de Informação) - FAI - Centro de Ensino Superior em Gestão, Tecnologia e Educação, Santa Rita do Sapucaí - MG, 2015.

ENDLER, D. **Dicas para Escrita de Casos de Teste**. Medium, 2018. Disponível em: <<https://medium.com/cwi-software/dicas-para-escrita-de-casos-de-teste-ccea14a7fdd9>>. Acesso em: 29 ago. 2019.

ESCOLA EM MOVIMENTO. **Escola em Movimento**. Belo Horizonte: [S. n.], 2016. Disponível em: <<https://www.escolaemmovimento.com.br/>>. Acesso em: 24 abr. 2019.

GOOGLE. Firebase. 2019. Disponível em: <<https://firebase.google.com/?hl=pt-BR>>.

JOHNSON, R E. Frameworks = (Components + Patterns). **Communications of the ACM**, v. 40, n. 10, oct. 1997.

KARNER, G. **Metrics for Objectory**. Thesis at the University of Linkopmg. Sweden, December, 1993.

KULPA, Cínthia Costa; PINHEIRO, Eluza Toledo; SILVA, Rogério Pierre. A Influência das Cores na Usabilidade de Interfaces Através do Design Centrado no Comportamento Cultural do Usuário. **Perspectivas em Gestão & Conhecimento** (PG&C): Universidade Federal da Paraíba: João Pessoa, 2011

LIBÂNEO, J. C. **Didática**. São Paulo: Editora Cortez, 1994.

MIRO. **Ensino Fundamental I**. Ribeirão Preto, SP, 2019. Disponível em: <<http://www.escolamiro.com.br/educacao-fundamental-i/aspectos-gerais/>>. Acesso em: 29 mai. 2019.

MORAN, J. M. **Novas tecnologias e mediação pedagógica**. 6. ed. Campinas: Papirus, 2000.

PRESSMAN, R. S. **Engenharia de Software**. 6. ed. São Paulo: McGraw Hill/Nacional, 2006.

PRESSMAN, R. S. **Engenharia de Software**. 7. ed. São Paulo: McGraw Hill/Nacional, 2011.

NIELSEN, J.; LORANGER, H. **Usabilidade na Web**: projetando websites com qualidade. Ed. Campus: Rio de Janeiro, 2007.

PROJECT MANAGEMENT INSTITUTE. **Conhecimento em Gerenciamento de Projetos Guia PMBOK**. 4. ed. Atlanta: PMI, 2008.

SANCHO, J. M. **Para uma tecnologia educacional**. 2. ed. Porto Alegre: Artmed, 2001.

SILBERSCHATZ, A. **Sistema de Banco de Dados**. 5a ed. Campus, 2006

SOMMERVILLE, I. **Engenharia de Software**: 9 ed. São Paulo: Pearson, 2011.

RODRIGUES, J. Y. **Entrevista**. Santa Rita do Sapucaí: [S. n.], 2019.

OBRAS CONSULTADAS

CYBIS, W.; BETIOL, A. H.; FAUST, R. **Ergonomia e usabilidade**: conhecimentos, métodos e aplicações. 2. ed. São Paulo: Novatec, 2010.

FAI - CENTRO DE ENSINO SUPERIOR EM GESTÃO TECNOLOGIA E EDUCAÇÃO. **Diretrizes para elaboração de trabalhos científicos**: padrão ABNT e adaptação às normas institucionais da FAI. Santa Rita do Sapucaí: FAI, 2018.

FREIRE, P. **Pedagogia da autonomia**: saberes necessários à prática educativa. 25. ed. São Paulo: Paz e Terra, 2000.

FREIRE, P. **Pedagogia do Oprimido**. Rio de Janeiro: Paz e terra, 1987.

ISO/IEC. **ISO/IEC FDIS 9126-1: Software Engineering – Product quality** – part 1: Quality Model. International Organization for Standardization, 2000.

NIELSEN, J. **Usability Engineering**. Boston: Academic Press, 1993.

PREECE, J. et al. **Human Computer Interaction**. Essex: Addison-Wesley, 1994.

PREECE, J.; ROGERS, Y.; SHARP, H. **Interaction design**: beyond human computer interaction. Chichester, UK: John Wiley and Sons Inc, 2002.

SECRETARIA DA EDUCAÇÃO, **Dia a Dia Educação**. Curitiba: Secretaria da Educação do Paraná, 2019

SHACKEL, B. **The concept of usability**. Englewood Cliffs, NJ: 1984. p. 45-87.

SHACKEL, B.; Usability – Context, framework, definition, design and evaluation. In SHACKEL, B; RICHARDSON, S. (eds.). **Human Factors for Informatics Usability**, Cambridge University Press, 1991. p. 21-38.

SANTOS, R. C. **Desenvolvimento de uma Metodologia para Avaliação de usabilidade de sistemas utilizando a lógica Fuzzy baseado na ISO**. 2007. 115 p.

SOCIEDADE BRASILEIRA DE COMPUTAÇÃO (SBC). **Ensino de Computação na Educação Básica**. SBC, 2019. Disponível em: <http://twixar.me/QllK>. Acesso em: 01 abr. 2019.

APÊNDICE A – EAP

A EAP do projeto está disponível na pasta APÊNDICE A do CD-ROM que acompanha este documento.

APÊNDICE B – DIAGRAMAS DE CASOS DE USO

Os diagramas de casos de uso do projeto estão disponíveis na pasta APÊNDICE B do CD-ROM que acompanha este documento.

APÊNDICE C – DIAGRAMA ENTIDADE RELACIONAMENTO

O Diagrama Entidade Relacionamento do projeto está disponível na pasta APÊNDICE C do CD-ROM que acompanha este documento.

APÊNDICE D – PLANILHA DOS RISCOS NEGATIVOS

A planilha dos riscos negativos encontra-se na pasta APÊNDICE D do CD-ROM que acompanha este documento.

APÊNDICE E – RELATÓRIO DE DESEMPENHO

O relatório de desempenho encontra-se na pasta APÊNDICE E do CD-ROM que acompanha este documento.

APÊNDICE F - MODELO INICIAL DA INTERFACE DE USUÁRIO

O modelo inicial da interface de usuário encontra-se na pasta APÊNDICE F do CD-ROM que acompanha este documento.

APÊNDICE G - DIAGRAMA DE PACOTES

O diagrama de pacotes encontra-se em anexo na pasta APÊNDICE G do CD-ROM que acompanha este documento.

APÊNDICE H - DIAGRAMA DE CLASSES

O diagrama de classes encontra-se em anexo na pasta APÊNDICE H do CD-ROM que acompanha este documento.

APÊNDICE I- DIAGRAMA DE OBJETOS

Os diagramas de objetos encontram-se na pasta APÊNDICE I do CD-ROM que acompanha este documento.

.

APÊNDICE J - DIAGRAMAS DE SEQUÊNCIA

Os diagramas de sequência encontram-se na pasta APÊNDICE J do CD-ROM que acompanha este documento.

.

APÊNDICE K – DESCRIÇÕES DOS CASOS DE USO

A descrição dos casos de uso encontra-se no APÊNDICE K do CD-ROM que acompanha este documento.

APÊNDICE L - DIAGRAMA DE VISÃO GERAL DA INTERAÇÃO

O diagrama de visão geral da interação encontra-se na pasta APÊNDICE L do CD-ROM que acompanha este documento.

.

APÊNDICE M - DIAGRAMA DE ATIVIDADE

O diagrama de atividade encontra-se na pasta APÊNDICE M do CD-ROM que acompanha este documento.

.

APÊNDICE N - MODELO LÓGICO

O modelo lógico encontra-se na pasta APÊNDICE N do CD-ROM que acompanha este documento.

.

APÊNDICE O - DICIONÁRIO DE DADOS DO MODELO LÓGICO

O dicionário de dados encontra-se na pasta APÊNDICE O do CD-ROM que acompanha este documento.

.

APÊNDICE P – PERFIL DO USUÁRIO

As entrevistas feitas com os futuros usuários encontram-se na pasta APÊNDICE P do CD-ROM que acompanha este documento.

.

APÊNDICE Q – LISTA DE VERIFICAÇÃO DA DOCUMENTAÇÃO

A lista de verificação da documentação feita pelo gerente de projeto encontra-se na pasta APÊNDICE Q do CD-ROM que acompanha este documento.

APÊNDICE R – CRONOGRAMA E DICIONÁRIO EAP

O Arquivo do *MS-Project* com o cronograma do projeto e o dicionário EAP encontra-se na pasta APÊNDICE R do CD-ROM que acompanha este documento.

APÊNDICE S – ESTIMATIVA DE ESFORÇO

O Arquivo do MS-Excel com a estimativa de esforço encontra-se no APÊNDICE S do CD-ROM que acompanha este documento.

APÊNDICE T – REQUISIÇÕES DE MUDANÇA

O modelo de requisições de mudança encontra-se no APÊNDICE T do CD-ROM que acompanha este documento.

APÊNDICE U – DIAGRAMA DE COMPONENTES

O arquivo com os diagramas de componentes encontra-se no APÊNDICE U do CD-ROM que acompanha este documento.

APÊNDICE V – ESTRUTURA FISICA DO BANCO DE DADOS

Os Arquivos com os *scripts* DDL, DML e de segurança do banco de dados encontram-se no APÊNDICE V do CD-ROM que acompanha este documento.

APÊNDICE W – RASTREABILIDADE ENTRE REQUISITOS E CASOS DE TESTE

Os requisitos funcionais a serem verificados encontram-se no APÊNDICE W do CD-ROM que acompanha este documento.

APÊNDICE X - DESCRIÇÃO DOS CASOS DE TESTE

A descrição dos casos de teste encontra-se no Apêndice X do CD-ROM que acompanha este documento.

APÊNDICE Y - HISTÓRICO DE REALIZAÇÃO

O histórico de realização de uma bateria dos casos de teste encontra-se no Apêndice Y do CD-ROM que acompanha este documento.

APÊNDICE Z – MANUAL DE USUÁRIO

Os manuais de usuário encontram-se no Apêndice Z do CD-ROM que acompanha este documento.

APÊNDICE AA – DIAGRAMA DE IMPLANTAÇÃO DA UML.

O Diagrama de implantação da UML encontra-se no Apêndice AA do CD-ROM que acompanha este documento.