

SiGeFA - Sistema de Gestão de Frequências de Alunos

Musekwa Evariste¹, Bruno Sousa^{1,2}

¹Universidade Aberta,

²CISUC-Centro de Informática e Sistemas da Universidade de Coimbra
1201887@estudante.uab.pt, bmsousa@dei.uc.pt

Resumo

A monitorização de assiduidade dos alunos nas salas de aulas é um desafio que os professores e os pais e encarregados de educação enfrentam passivamente nas escolas secundárias moçambicanas. O presente artigo apresenta a implementação de um Sistema de Gestão de Frequências de Alunos (SiGeFA) composto por uma aplicação móvel, uma aplicação web e aplicação backend que visa monitorizar a assiduidade dos alunos por meio de marcação de faltas dos alunos ausentes e de envio automático de notificações aos pais e encarregados de educação cujos alunos estiverem ausentes durante as aulas. A avaliação do primeiro protótipo aponta para uma aceitação deste sistema por parte de professores e alunos.

Palavras-chave: assiduidade, gestão de frequências, aplicação móvel.

Title: SiGeFA – A Student Frequency Management System

Abstract

An effective monitoring solution of student attendance in the classroom is a challenge that teachers, parents and educators face passively in the secondary schools of Mozambique. This paper introduces the implementation of A Student Frequency Management System (SiGeFA) prototype composed by a mobile application, a web application and a backend application, which aims to monitor the attendance of students in the classroom. Through the mobile application teachers can manage the attendance of students, registering the student's absences and notifying automatically the parents and educators regarding the non-attendance of their own sons/daughters in the classes. The evaluation of the first prototype puts in evidence the acceptance of SiGeFA by teachers and students in a Mozambique school.

Keywords: student attendance, management of attendance, mobile application.

1. Introdução

A assiduidade dos alunos nas salas de aulas constitui uma preocupação para professores, pais e encarregados de educação da escola secundária Januário Pedro onde um dos autores participa nas atividades de leccionamento. Diversos autores [Menezes, 2013; Maria Ivone, 2017] referem que o problema da assiduidade é comum em diversas escolas secundárias de Moçambique. Os alunos tendem a perder o interesse em ser assíduos às aulas. Estes faltam propositadamente às aulas mesmo quando estão dentro do recinto escolar. Ademais, não existe nenhum mecanismo para os pais monitorizarem, à distância, a participação dos seus educandos nas aulas, nem mesmo mecanismos para os professores notificarem os pais/encarregados de educação sempre que os seus educandos se ausentam às aulas. Neste contexto os encarregados de educação não têm conhecimento deste desinteresse dos seus educandos, não podendo atuar para reverter as faltas.

As tecnologias de informação têm sido utilizadas para fomentar a aprendizagem de alunos dentro e fora da sala de aulas, desde o ensino básico [Marta, 2016] ao ensino superior [Maria Ivone, 2017]. Noutra vertente, e diretamente relacionada com a problemática da assiduidade, algumas abordagens privilegiam a utilização de sensores para o controlo da assiduidade de alunos ou até mesmo do pessoal administrativo das escolas [Abuarqoub, 2017]. Estes sistemas de registo da assiduidade – *Automated Attendance Systems (AAS)* – têm tido bastante interesse, seja no sentido de reduzir o esforço necessário para a marcação de faltas de alunos na sala de aula [Sanja, 2016] utilizando aplicações móveis instaladas nos *smartphones* dos alunos, seja através de técnicas de inteligência artificial para o reconhecimento facial de alunos a partir de fotografias.

Tendo presente a realidade da fraca assiduidade e as suas consequências no processo de ensino e aprendizagem, projetou-se e validou-se um sistema de gestão de frequências de alunos – SiGeFA para a escola secundária Januário Pedro. O SiGeFA simplifica o processo de controlo da assiduidade dos alunos por parte dos professores e promove o envolvimento dos encarregados de educação no processo de educação dos seus educandos ao fornecer, em tempo real informação das faltas dos seus educandos. O SiGeFA é um sistema que visa motivar os alunos a participar nas aulas, ao permitir o acesso a um conjunto de estatísticas alusivas à assiduidade, bem como a apresentação de mensagens a promover a participação dos alunos numa dada aula.

A arquitetura do SiGeFA, apresentada neste artigo inclui diversas componentes, uma componente servidora que pode ser instalada em contexto *cloud* ou na própria infraestrutura da escola, com acesso à base de dados para registo das faltas. A componente móvel permite a interação com os diversos utilizadores do sistema, incluindo professores, alunos e encarregados de educação, onde cada um tem acesso a funcionalidades distintas, consoante o tipo de utilizador. O SiGeFA suporta a administração da informação através de uma aplicação *web*.

A avaliação do protótipo do SiGeFA por parte de professores da escola secundária Januário Pedro aponta para uma boa aceitação do sistema, sendo unanime a contribuição do SiGeFA para a melhoria da assiduidade dos alunos nas salas de aulas.

A organização deste artigo é a seguinte. A secção 2 descreve trabalho relevante relacionado. A secção 3 dedica-se à descrição do protótipo do sistema. A secção 4 apresenta a avaliação do protótipo e a secção 5 faz algumas considerações sobre a aplicação final, expondo também ideias de trabalho futuro.

2. Trabalho Relacionado

As tecnologias de informação têm introduzido inovações em diversos setores como a educação [Maria Ivone, 2017]. Desde o ensino básico [Marta, 2016] ao ensino superior que as tecnologias de informação têm sido empregues para fomentar a aprendizagem dos alunos dentro e fora da sala de aulas. A utilização das aplicações móveis assume particular interesse, na promoção de pensamento crítico e de práticas para ensino colaborativo [Houston, 2017].

Diversos estudos defendem o uso integrado de diversas tecnologias incluindo sensores, (IoT - *Internet of Things*) quer para controlo da assiduidade de alunos, de pessoal administrativo quer para a gestão eficiente de edifícios (ex: campus universitário) em termos de utilização da energia e da água [Abuarqoub, 2017].

Muffin [Sanja, 2016] é um sistema que visa facilitar o registo da assiduidade de alunos no contexto da sala de aula, com o objetivo de reduzir o tempo do professor para a marcação de faltas de alunos. Este sistema consiste numa aplicação móvel instalada no *smartphone* dos alunos e num dispositivo na sala de aula baseado na plataforma Arduino [Arduino, 2019] que recebe informação da presença dos alunos. À semelhança do SiGeFA o Muffin possui também uma componente servidora que dá acesso a um conjunto de estatísticas da assiduidade dos alunos nas diversas classes, contudo o Muffin não promove a relação entre escola e encarregados de educação, um fator relevante em contexto de ensino primário e secundário.

Numa vertente de possibilitar mecanismos automáticos de registo da assiduidade – *Automated Attendance Systems* (AAS) – alguns autores exploram o uso de técnicas de inteligência artificial para possibilitar o reconhecimento facial de alunos [Samuel, 2016]. Neste tipo de abordagem o professor tira fotografia(s) da sala de aula incluindo a face de todos os alunos e a partir destas fotografias o sistema identifica os alunos possibilitando depois o registo da sua assiduidade. Contudo este tipo de solução pode ter alguma imprecisão, dado que podem não ser identificados todos os alunos e requer treino dos algoritmos com fotografias individuais dos alunos.

Um fator comum entre os diversos estudos e o SiGeFA prende-se com a falta de motivação dos alunos para o processo de aprendizagem, verificando-se baixos níveis de assiduidade por parte de alguns alunos e de diversas perturbações na sala de aula quando presentes.

3. SiGeFA

O Sistema de Gestão Frequências de Alunos (SiGeFA) foi desenhado e desenvolvido, por um lado, para simplificar o processo de obtenção de informação estatística sobre o nível de

assiduidade dos alunos nas escolas moçambicanas e, por outro lado, para promover um envolvimento ativo por parte dos encarregados de educação no processo de educação dos seus educandos. A especificação e desenvolvimento do SiGeFA baseou-se em metodologias *Agile* [Luís Gonçalves, 2019], para permitir a disponibilização de um protótipo funcional para avaliação e recolha de feedback por professores da escola secundária Januário Pedro.

A arquitetura do SiGeFA inclui diversos componentes: 1) aplicação web que permite à administração de qualquer escola introduzir os dados dos alunos e professores no sistema; 2) aplicação móvel que permite aos professores efetuarem o registo da assiduidade dos alunos e aos encarregados de educação fossem receberem de forma instantânea, notificações das faltas dos seus educandos; 3) uma componente servidora que disponibiliza um conjunto de interfaces para permitir a informação em base de dados.

3.1 Arquitetura

O desenho da arquitetura do SiGeFA considerou requisitos de alta disponibilidade, de performance e de acessibilidade, nomeadamente para a componente servidora funcionar em plataformas de alta disponibilidade (ex: *cloud*), a componente móvel ficar acessível aos utilizadores nos seus próprios *smartphones* e uma componente web para administração da plataforma SiGeFA, fácil de utilizar. O critério de performance foi assegurado através do desenho de interfaces para comunicação entre os diversos componentes. Por exemplo, a comunicação com a componente servidora recorrer a APIs *Representational State Transfer* - REST que permitem a troca de dados em formato *JavaScript Object Notation* - JSON [IBM, 2018]. As aplicações baseadas na tecnologia REST utilizam os pedidos definidos no protocolo *Hypertext Transfer Protocol* – HTTP [Wikipedia – HTTP, 2019], amplamente utilizado na *Internet*.

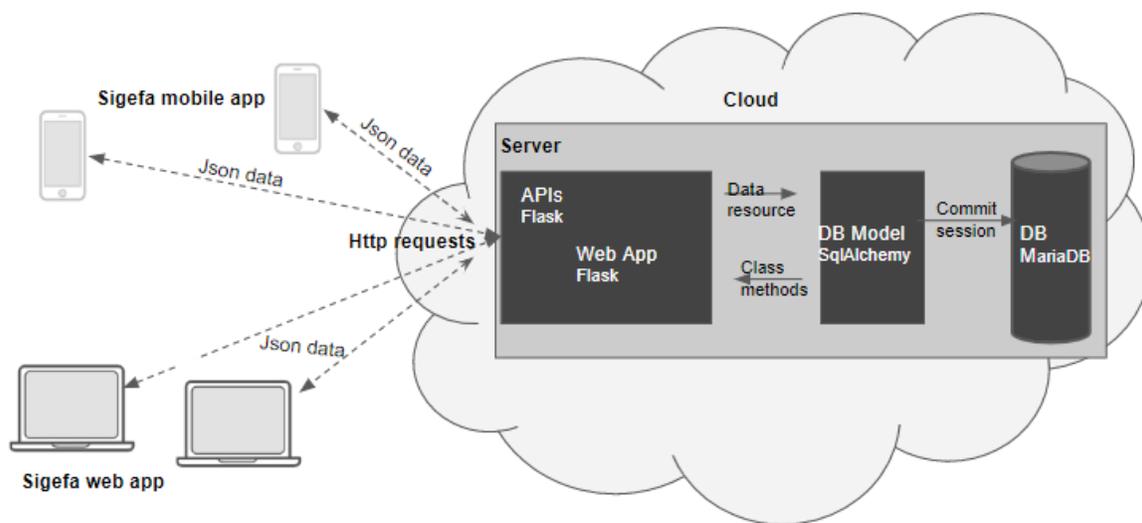


Figura 1. Arquitetura SiGeFA.

A Figura 1 apresenta uma instanciação da arquitetura do SiGeFA, evidenciando os diversos componentes, a aplicação móvel, a aplicação web e a componente servidora. A componente servidora inclui aplicações que disponibilizam interfaces *REST* para a troca de dados e para gerir a comunicação com a base de dados. Uma instanciação desta arquitetura inclui a implementação do servidor recorrendo a tecnologia *Python/Flask* [Flask, 2019] que permite a implementação de interfaces *REST* de uma forma escalável e recorrendo à Framework *Python/SQLAlchemy* [SQLAlchemy, 2019] para abstração dos modelos de dados. A nível da base de dados recorreu-se ao *MariaDB* [MariaDB, 2019] um motor de base de dados de código-fonte aberto.

3.2 Aplicação Móvel

A aplicação móvel serve essencialmente de interface para o utilizador, tendo sido desenhada para suportar diferentes tipos de utilizadores: professor, aluno e pai. Todo o conteúdo nos diversos ecrãs é disponibilizado em função do tipo de utilizador. O processo de registo no sistema é feito através da aplicação em que cada utilizador indica a escola em que os alunos estão associados, indica a conta de acesso pretendida e as credenciais a utilizar para futuros acessos. O processo de registo é validado pela administração de uma escola. A Figura 2 evidencia o fluxograma da informação na aplicação móvel, onde é possível verificar as regras que se aplicam aos diversos perfis de utilizador. Por exemplo, só os professores é que conseguem marcar faltas.

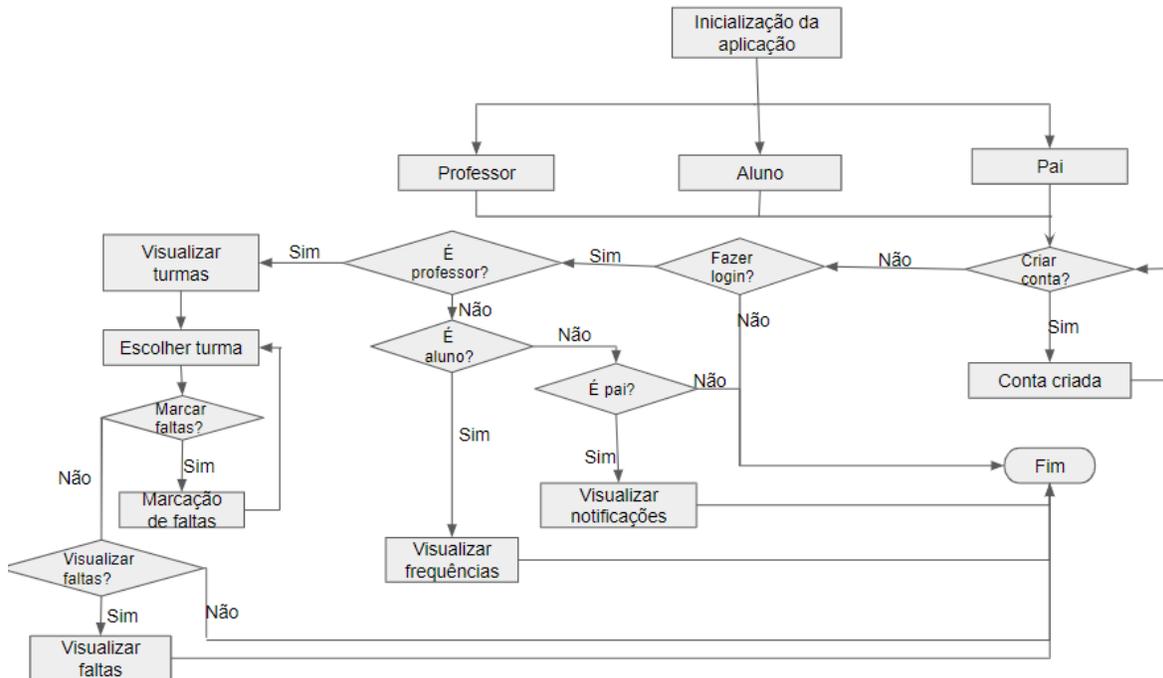


Figura 2. Fluxograma da aplicação móvel.

Para além do fluxo de informação na aplicação, também foram elaborados os esboços da interface dos diversos ecrãs da aplicação móvel, no sentido de garantir que os requisitos e

são validados tendo em conta o fluxo de informação e os diversos tipos de utilizadores. A Figura 3 e a Figura 4 demonstram exemplos dos esboços da interface do ecrã inicial e do ecrã de assiduidade.

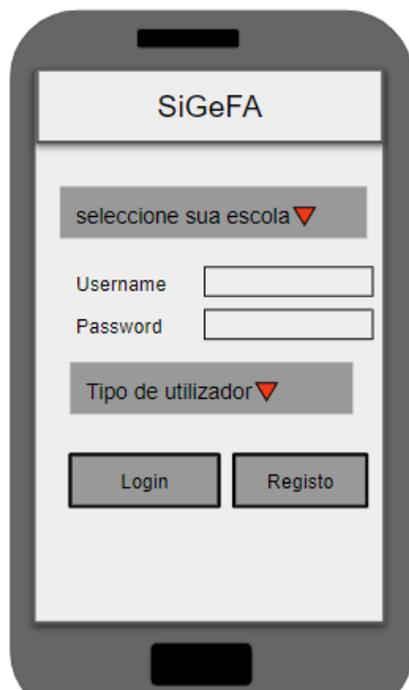


Figura 3. Esboço do ecrã inicial

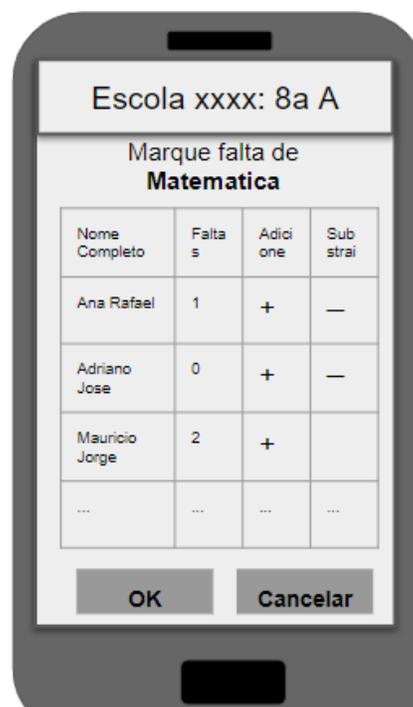


Figura 4. Esboço do ecrã de assiduidade

Como já mencionado anteriormente um dos critérios tidos em conta no desenho do sistema, foi o da acessibilidade, neste sentido privilegiou-se a utilização de *frameworks* que permitem a elaboração de aplicações móveis em diversos sistemas operativos móveis (Google - Android, Apple - iOS). A implementação da aplicação móvel foi feita em React-Native [React-Native, 2019] que possibilita o desenvolvimento de aplicações de uma forma simples e é suportada por diversas plataformas que possibilitam o desenvolvimento de protótipos de uma forma integrada. A plataforma Expo [Expo, 2019] foi utilizada para permitir o teste da aplicação móvel e a sua comunicação com a componente servidora.

O fluxograma da aplicação validado nos esboços de interfaces, conduziu à implementação dos diversos ecrãs da aplicação. A Figura 5 e Figura 6 exemplificam dois dos ecrãs desenvolvidos como protótipo desta aplicação. Um destes ecrãs é o de inicialização (Figura 5) da aplicação, que para utilizar o SiGeFA pela primeira vez, deve antes de tudo, registar-se. A operação de registo requer que o novo utilizador navegue para o ecrã de registo, selecionando a escola e o tipo de perfil/utilizador. O registo é concluído quando o utilizador clica no botão de “Registo”. Ao clicar neste botão, o utilizador é redirecionado para o ecrã definido no fluxograma: Se tiver indicado “Aluno”, como tipo de utilizador, então é direcionado para o ecrã de registo como aluno; caso seja professor ou pai/encarregado de educação, é direcionado para o ecrã respetivo ecrã. Caso o utilizador já se tenha sido registado, ele precisa de fazer autenticação para poder ter acesso à sua conta. No processo

de aceder à sua conta, o utilizador deve indicar a escola, o seu nome de utilizador, a sua palavra-passe e o tipo de utilizador.



Figura 5. Protótipo ecrã inicial

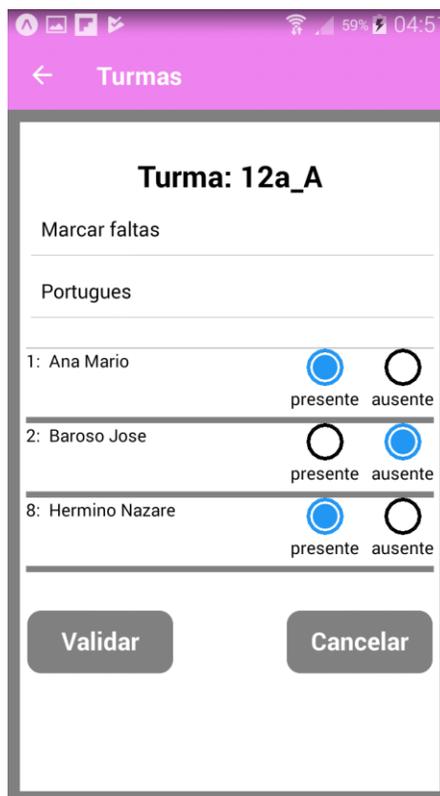


Figura 6. Protótipo ecrã de assiduidade

Uma das funcionalidades mais importantes do SiGeFA é o controlo de assiduidade. A Figura 6 demonstra o ecrã de controle de assiduidade, em que um professor pode marcar as presenças ou faltas dos alunos de uma turma e de uma disciplina, após ter escolhido a turma e a disciplina que leciona. Dada a importância desta funcionalidade, são exemplificados alguns trechos de códigos-fonte escritos no âmbito da implementação desta aplicação, como se ilustra na Figura 7 e Figura 8 para a atualização da assiduidade e para a validação de frequências por turma.

```
//api: PUT => método para a atualização de frequências (faltas/presenças)
const putFrequencias = async (params1, params2)=>{
  try{
    let response = await fetch(params1, {
      method: 'PUT',
      headers: {
        'Accept': 'application/json',
        'Content-Type': 'application/json',
      },
      body: JSON.stringify(params2)
    });
    let responseJson = await response.json()
    return responseJson.result;
  }
  catch(error){
    return (`Error is : ${error}`);
  }
}
```

Figura 7. Método de atualização da assiduidade

```
// método para validação de frequências(faltas/presenças)
validarFaltas = ()=>{
  const url ='http://52.47.208.23/escolas/frequencias/'+this.state.turmalid;
  console.log(this.state.sessaoDeFaltas)
  const items = this.state.sessaoDeFaltas
  for(var i=0; i<items.length; i++){
    putFrequencias(url, items[i])
  }
}
```

Figura 8. Método de validação de frequências

3.3 Aplicação Servidora (*backend*)

A componente servidora do SiGeFA foi desenvolvida recorrendo à linguagem Python 3.2 ["Python 3.2", 2011] e foi concebido e desenhado de modo a garantir requisitos de alta disponibilidade e de performance, por exemplo para suportar múltiplos pedidos simultâneos de aplicações móveis e de aplicações web.

A componente servidora do SiGeFA implementa uma arquitetura do tipo cliente-servidor, na qual um cliente comunica com o servidor através de uma API REST bem definida. Esta API foi definida e implementada de acordo com as operações a efetuar. Por exemplo, o cliente desencadeia o pedido recorrendo aos métodos (GET, POST, PUT, DELETE, entre outros) do protocolo HTTP e associados a opções/operações específicas da API do

servidor. Por sua vez, o servidor ao receber um pedido valida o mesmo, garantindo que os campos obrigatórios foram preenchidos com valores corretos por parte do cliente. Após validação, e consoante o tipo de operação o servidor pode criar um recurso para acesso (de escrita, e/ou de leitura) à base de dados, retornando a resposta ao cliente em formato JSON.

O trecho de código exemplificado na Figura 9 para a componente servidora demonstra a utilização da biblioteca *Flask* do *Python* e da biblioteca *JSON Web Tokens (JWT)* [JWT, 2019] para a geração de *tokens* no sentido de tornar a comunicação com o servidor mais segura.

```
from flask import Flask, jsonify
from flask_restful import Api
from flask_sqlalchemy import SQLAlchemy
from flask_jwt_extended import JWTManager
from flask_migrate import Migrate

app = Flask(__name__)
api = Api(app)

from db.models import db, RevokedTokenModel
import resources

db.init_app(app)
jwt = JWTManager(app)
migrate = Migrate(app, db)

@app.before_first_request
def create_tables():
    db.create_all()

if __name__ == '__main__':
    app.run(debug=True)
```

Figura 9. Código da componente servidora.

Como já mencionado anteriormente, são utilizadas diversas bibliotecas para o desenvolvimento da componente servidora do SiGeFA. A biblioteca *SQLAlchemy* [SQLAlchemy, 2019] permite fazer toda a interação com a base de dados, desde a atualização da informação até a uma simples consulta. O modelo de dados, pode ser sincronizado de forma simples com a base de dados e a própria aplicação através da utilização da biblioteca *Flask-Migrate* [Flask-Migrate, 2019]. As APIs em REST no servidor são implementadas na aplicação através da utilização da biblioteca *Flask*, que funcionando como uma aplicação web, permite definir as opções corretas para as diversas operações. Por exemplo a operação de marcação de faltas, exemplificada na seção anterior requer a disponibilização da API no endereço */escolas/frequências/id_turma* e a passagem de informação é feita através do verbo PUT do protocolo HTTP.

3.4 Aplicação Web de Administração

A componente Web do SiGeFA permite fazer a administração dos dados no sistema, desde o registo de uma escola, das turmas, das diversas disciplinas e dos respetivos professores. A Figura 10 demonstra o fluxograma associado à aplicação Web, nomeadamente à operação de adicionar informação e das respetivas validações necessárias. Para tal consideraram-se cinco formulários que permitem a inserção, leitura, atualização e remoção de dados da escola, dos professores, dos alunos, das disciplinas e das turmas.

Conforme já mencionado, um dos requisitos do SiGeFA é funcionar em múltiplos sistemas operativos, assim para a elaboração do protótipo da aplicação Web utilizou-se *React-Native* [React, 2019], uma biblioteca de *JavaScript* vocacionada à criação de interfaces de utilizador.

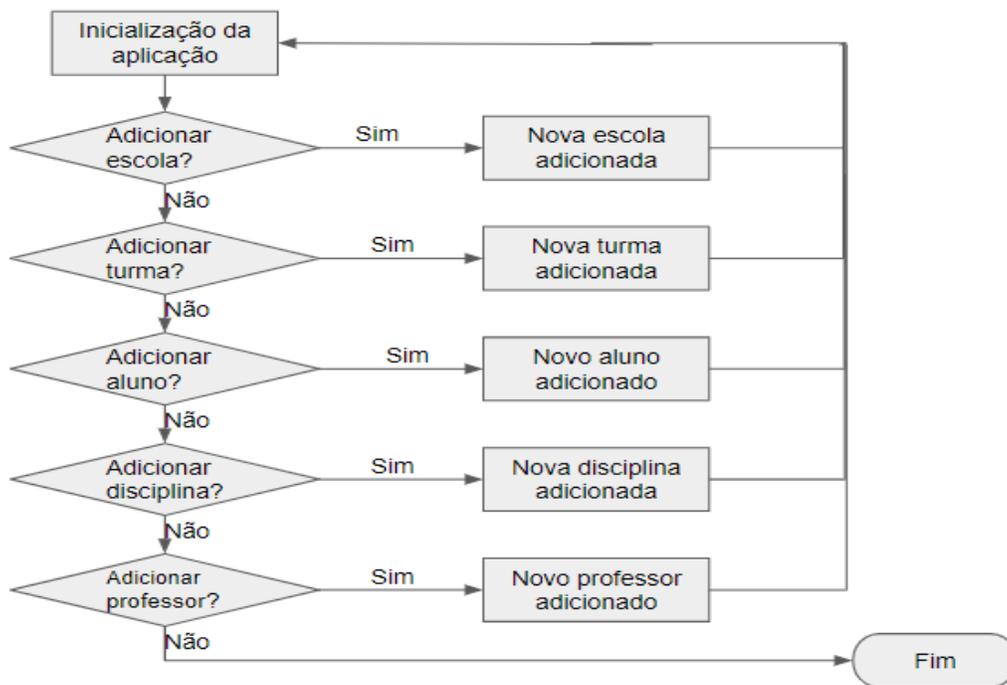
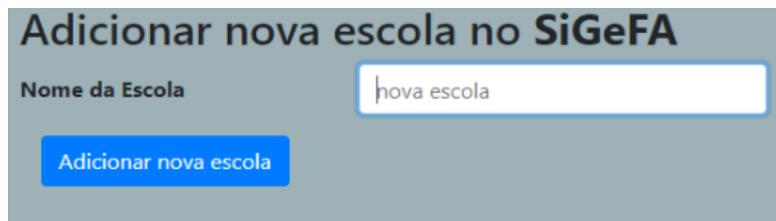


Figura 10. Fluxograma da aplicação web de administração

O fluxograma ilustrado na Figura 10 descreve apenas o processo de registo, inserção de dados das diferentes entidades no sistema (escola, aluno, etc.). Para cada uma das entidades foram criados formulários com as respetivas regras de validação, por exemplo na inserção de uma escola deve ser verificado se a mesma já está registada no SiGeFA.



Adicionar nova escola no SiGeFA

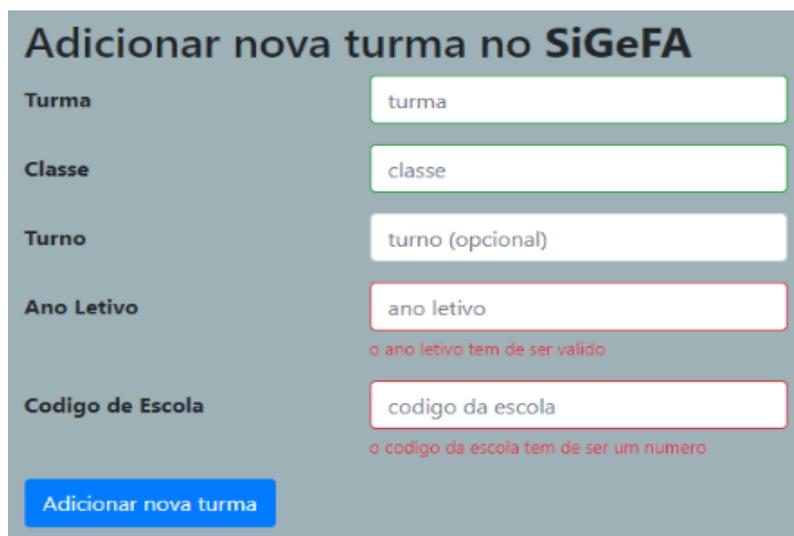
Nome da Escola

Adicionar nova escola

Figura 11. Formulário de adição de uma nova escola

A Figura 11 ilustra o formulário com um campo de entrada para o registo de uma nova escola. Neste formulário deve ser indicado o nome da escola para ser registado no SiGeFA.

Para a criação de uma nova turma a ser associada a uma dada escola já existente, o utilizador do sistema (ex: administração da escola) deve preencher os campos de entrada de dados da Figura 12. especificando a turma, a classe, o turno, o ano letivo e o código da respetiva escola.



Adicionar nova turma no SiGeFA

Turma

Classe

Turno

Ano Letivo
o ano letivo tem de ser valido

Codigo de Escola
o codigo da escola tem de ser um numero

Adicionar nova turma

Figura 12. Formulário de adição de uma nova turma

A Figura 13 exemplifica um trecho de código em *React-Native* associado à implementação do formulário para o registo de uma nova escola no SiGeFA.

```
class Escola extends Component{
render(){
  return(
    <div className='row row-content'>
      <div className='col-12'>
        <Form onSubmit={this.handleSubmit}>
          <FormGroup row>
            <Label htmlFor='escola' md={3}><strong>Nome da Escola</strong></Label>
            <Col md={4}>
              <Input type='text' id='escola' name='escola'
                placeholder='nova escola' value={this.state.escola}
                onChange ={this.handleInputChange}
              />
            </Col>
          </FormGroup>
          <FormGroup>
            <Col md={{size: 8}}>
              <Button type='submit' color='primary'>Adicionar nova escola</Button>
            </Col>
          </FormGroup>
        </Form>
      </div>
    </div> ) }
export {Escola}
```

Figura 13. Código para adicionar uma nova escola

4. Avaliação do Protótipo

Foi elaborado um protótipo do SiGeFA a fim de ser validado por professores da escola secundária Januário Pedro. O tipo de avaliação incidiu sobretudo na satisfação dos utilizadores considerando a perspetiva do perfil do professor. A componente servidora do protótipo do SiGeFA foi instalada na *Amazon Web Services Cloud Computing Services* [Amazon AWS, 2019], no sentido de replicar o cenário real de instalação do SiGeFA, em que está disponível via Internet para possibilitar aos encarregados de educação a monitorização da assiduidade dos seus educandos.

Para se aferir o grau de satisfação dos professores potenciais utilizadores da aplicação do SiGeFA, realizou-se, por meio de um questionário, uma avaliação. Foram aleatoriamente escolhidos cinco professores da escola secundária Januário Pedro para uma participação voluntária na avaliação do protótipo. Cinco professores que lecionam cinco disciplinas diferentes (Português, História, Francês, Matemática e Química), dos quais três homens e duas mulheres.

A aplicação móvel foi disponibilizada aos professores através de um código QR, em que eram dadas instruções dos objetivos da aplicação e do que se pretendia avaliar. Os professores que participaram no estudo de avaliação procederam à instalação da *framework Expo* e da aplicação nos seus respetivos telemóveis, tendo sido convidados a realizar as

operações de registo da assiduidade dos alunos através do SiGeFA, previamente carregado com os dados das disciplinas e turmas destes professores.

Após a experimentação que durou duas semanas, foi disponibilizado um questionário no sentido de recolher o feedback dos professores quanto ao seu nível de satisfação.

4.1 Questionários a potenciais utilizadores

Dada a participação voluntária por parte dos professores, foi elaborado um questionário simples mas com incentivos à partilha de feedback para a melhoria do SiGeFA e das suas aplicações. Foram consideradas cinco questões no questionário:

1. Este sistema contribui para a melhoria da assiduidade às aulas?
2. A aplicação é fácil de utilizar, intuitiva e compreensível?
3. A aplicação funciona de modo correto?
4. A aplicação garante a segurança de dados dos utilizadores?

O questionário considerou para as diversas questões uma escala linear em que 1- representa *Não concordo* e 5- corresponde a *Concordo plenamente*. Nesta escala temos os seguintes valores: 1- *Não concordo plenamente*, 2- *Não Concordo mas*, 3- *Concordo tendencialmente*, 4- *Concordo com algumas reservas* e 5- *Concordo plenamente*.

4.2 Resultados da avaliação

Os resultados obtidos nesta avaliação do protótipo do SiGeFA foram encorajadores. Todos os cinco professores submetidos ao questionário foram unânimes e concordaram plenamente que o SiGeFA contribui para a melhoria da assiduidade às aulas e para uma maior aproximação entre os encarregados de educação e a escola. Os professores também concordaram plenamente que o SiGeFA é fácil de utilizar, intuitivo e compreensível, e que a aplicação móvel disponibilizada no protótipo funciona corretamente.

Mas, no que se refere à segurança de dados dos utilizadores, todos os professores submetidos a este questionário também foram unânimes: a aplicação requer melhorias no sentido de garantir a segurança dos dados dos seus utilizadores. Por exemplo, as interfaces da componente servidora devem ser disponibilizadas em modo seguro (i.e. HTTPS). Por um lado, verificou-se que a aplicação não restringia a visualização de dados, isto é, qualquer utilizador podia ter acesso ao registo de faltas de qualquer aluno desde que conhecesse a turma e o número desse aluno. Por outro lado, não existia mecanismo de autenticação que permitisse verificar se o utilizador que se fizesse passar por professor era realmente professor ou não. Também não existia mecanismo de restringir que um dado professor não pudesse marcar faltas aos alunos que não ensinasse e na disciplina que não lecionasse.

A Figura 14 demonstra o resultado obtido na questão “Este sistema contribui para a melhoria da assiduidade às aulas?”. Podendo-se verificar a excelente aceitação por parte dos professores que participaram na avaliação do SiGeFA.

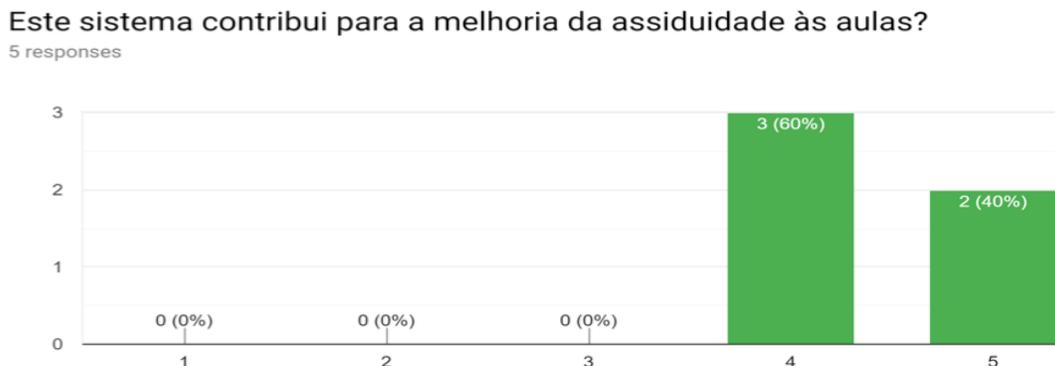


Figura 14. Resultado do questionário de avaliação e de contribuições do SiGeFA.

5. Conclusões e trabalho futuro

O SiGeFA, descrito neste artigo, visa reduzir a fraca assiduidade de alunos em escolas Moçambicanas, ao promover a comunicação entre a escola e os encarregados de educação, ao permitir o envio de notificações quando um aluno falta a uma dada aula. O SiGeFA atua para além de um sistema de controlo de faltas, dado que possibilita a informação da assiduidade de um aluno aos encarregados de educação em tempo real e inclui aspetos motivacionais para combater as faltas às aulas.

Os resultados da avaliação do protótipo do SiGeFA na escola secundária Januário Pedro em Moçambique demonstram que este tipo de sistema pode efetivamente contribuir para uma gestão mais eficiente da assiduidade dos estudantes e melhorar a comunicação entre os encarregados de educação e a escola. Diversos professores que participaram na avaliação do primeiro protótipo apontam o SiGeFA como um sistema que já devia estar em funcionamento.

Os próximos passos visam a implementação do SiGeFA na escola secundária Januário Pedro, envolvendo todos os intervenientes desde alunos, professores e encarregados de educação. Pretende-se também conduzir uma avaliação de larga escala, no sentido de expandir o SiGeFA a outras escolas de Moçambique com a mesma problemática da assiduidade dos alunos.

Referências

Menezes, Leonarda Jacinto José Maria, "O ensino bilíngue em Moçambique: entre a casa e a escola", 2013.

Maria Ivone Cardoso, Ana Paula Camarinha, "A revolução das TIC: desafio da educação no século XXI", FORGES, 2017, <http://recipp.ipp.pt/handle/10400.22/10674>.

Marta Gómez Domingo, Antoni Badia Garganté, "Exploring the use of educational technology in primary education: Teachers' perception of mobile technology learning impacts and applications' use in the classroom", Computers in Human Behavior, Volume 56, 2016, Pages 21-28, ISSN 0747-5632, <https://doi.org/10.1016/j.chb.2015.11.023>.

Abuarqoub, Abdelrahman et al., "A Survey on Internet of Things Enabled Smart Campus Applications", Proceedings of the International Conference on Future Networks and Distributed Systems, ISBN 978-1-4503-4844-7, ACM, <https://doi.org/10.1145/3102304.3109810>.

Houston Heflin, Jennifer Shewmaker, Jessica Nguyen, "Impact of mobile technology on student attitudes, engagement, and learning", Computers & Education, Volume 107, 2017, Pages 91-99, ISSN 0360-1315, <https://doi.org/10.1016/j.compedu.2017.01.006>.

Sanja Maravic Čisar, R. Pinter, V. Vojnić, V. Tumbas and P. Čisar, "Smartphone application for tracking students' class attendance," 2016 IEEE 14th International Symposium on Intelligent Systems and Informatics (SISY), Subotica, 2016, pp. 227-232. <https://doi.org/10.1109/SISY.2016.7601502>.

Samuel Lukas, A. R. Mitra, R. I. Desanti and D. Krisnadi, "Student attendance system in classroom using face recognition technique," 2016 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, 2016, pp. 1032-1035. <https://doi.org/10.1109/ICTC.2016.7763360>.

Python 3.2 Release, <https://www.python.org/download/releases/3.2/> (2011) [11 de Abril de 2019].

Python Flask, <http://flask.pocoo.org> (2019) [11 Abril de 2019].

Flask-Migrate, <https://flask-migrate.readthedocs.io/en/latest/>, (2019) [11 Abril de 2019].

Python SQLAlchemy, <https://www.sqlalchemy.org> (2019) [11 Abril de 2019].

MariaDB, <https://mariadb.org> (2019) [11 Abril de 2019].

React-Native, <https://facebook.github.io/react-native/> (2019) [11 de Abril de 2019].

Expo, <https://expo.io> (2019) [11 de Abril de 2019].

JWT, JSON Web Tokens, <https://jwt.io> (2019) [11 de Abril de 2019].

Arduino, <https://www.arduino.cc> (2019) [25 de Maio de 2019].

Luís Gonçalves, Metodologia Agile, Tudo o que precisa de saber sobre este tema, <https://luis-goncalves.com/pt-pt/o-que-e-metodologia-agile/> (2019) [25 de Maio de 2019].

IBM, REST in the cloud, <https://developer.ibm.com/articles/cl-restfulapisincloud/> (2018) [25 de Maio de 2019].

Wikipédia Hypertext Transfer Protocol - HTTP, https://pt.wikipedia.org/wiki/Hypertext_Transfer_Protocol (2019) [25 de Maio de 2019].

Amazon Web Services – AWS, <https://aws.amazon.com> (2019) [25 de Maio de 2019].



Musekwa Iguna Evariste, Licenciado em Engenharia Informática pela Universidade Aberta do Portugal em 2018 e em Gestão e Administração Educacional pela Universidade Católica de Moçambique em 2010. Professor do ensino secundário em Moçambique desde 2010. Tem como áreas de interesse o desenvolvimento de aplicações web e móveis com recurso às tecnologias Flask/Python, SQLAlchemy/Python, MySQL, React/JavaScript, React Native/JavaScript e Node/JavaScript.



Bruno Sousa é investigador do CISUC—Centro de Investigação e Sistemas da Universidade de Coimbra, é autor de diversas publicações em revistas, conferências e workshops. Os seus interesses de investigação incluem resiliência de serviços/aplicações em ambientes cloud, segurança de serviços/aplicações e de IoT. Tem uma vasta experiência em projetos de investigação, incluindo projetos financiados por fundos FP6, FP7, H2020, CATRENE e fundos nacionais FCT e P2020. Presentemente, é professor auxiliar convidado no departamento de engenharia informática da FCTUC