

Software Requirements Specification

Gustavo Meireles, José Pombo, Vanessa Lai Leong, Tomás Costa

September 25, 2023

1 Preamble

Este documento é destinado aos desenvolvedores e clientes da aplicação em questão. Esta é a primeira versão deste documento, estando a ser escrita para a disciplina Engenharia de Software.

Esta aplicação será utilizada para permitir o envio, recebimento e avaliação de trabalhos entre alunos e professores. Estas três funcionalidades são as principais funções da aplicação, sendo necessárias para a sua utilidade básica.

2 Introduction

A aplicação permite a um aluno enviar trabalhos a um professor em específico. O aluno será avisado quando o trabalho em questão é recebido pelo professor a quem o trabalho foi enviado. Este será capaz de avaliar o trabalho enviado pelo aluno e enviar a avaliação ao aluno em questão. Sempre que um aluno quiser enviar um documento a um professor, o aluno irá fazer o upload do documento para um servidor central que irá consequentemente enviar o mesmo documento para o professor especificado pelo aluno. Depois de o professor receber o documento, uma notificação é enviada ao aluno a dizer que o documento foi recebido com sucesso.

Depois de realizar a sua avaliação, o professor pode mandar a mesma ao aluno, esta avaliação será enviada como um email a partir do servidor principal. Este email irá conter a avaliação do professor e informará qual o documento avaliado. Depois do envio da avaliação pelo professor e o recebimento da mesma pelo aluno, o documento guardado no servidor será apagado. Todo este processo é o ciclo principal da aplicação, sendo repetido para todos os envios de trabalhos. O processo descrito deve ser facilmente realizado com um computador e qualquer ficheiro em formato pdf ou qualquer outro formato guardado em pleno texto (pdf é o formato preferível para evitar qualquer possibilidade de acontecerem alterações ao documento durante o seu tempo de vida dentro do sistema da aplicação).

Contents

1	Preamble	1
2	Introduction	1
3	Functional Requirements	3
3.1	User authentication	4
3.1.1	User registration	4
3.1.2	User login	5
3.2	Submission of work	6
3.3	Storing of work	7
3.3.1	Online storing	7
3.3.2	Offline storing and backups	7
3.4	Evaluation of Work	8
3.5	Notifications	9
3.5.1	Notifications for students	9
3.5.2	Notifications for professors	10
4	Non-Functional Requirements	11
4.1	Performance	11
4.2	Security	11
4.3	Usability	11
4.3.1	Interação gráfica	12
4.3.2	Interação por linha de comandos	12
4.4	Scalability	12
4.5	Stability	12
4.6	Notification Delivery	12

3 Functional Requirements

- User authentication;
 - User registration;
 - User login;
- Submission of work;
- Receiving of work;
- Storing of work:
 - Offline storing of work;
 - Online storing of work;
 - Backup of work
- File tampering verification;
- Evaluation of work;
- User notifications;
 - Professor notifications;
 - Student notifications;

3.1 User authentication

O sistema deve fornecer autenticação de utilizadores para que os professores e alunos possam aceder às suas contas de forma segura. Os utilizadores criados e existentes serão também utilizados para o funcionamento da aplicação, existindo permissões e proibições diferentes entre uma conta de aluno e uma conta de professor.

3.1.1 User registration

A plataforma não terá qualquer conta inicialmente. Os alunos e professores terão de criar as suas contas para poderem interagir com qualquer parte da aplicação. Consequentemente, é necessário ser possível realizar o registo de utilizadores.

O registo de utilizadores necessita do fornecimento de vários dados distintos:

- posição:
 - professor;
 - * turmas
 - * escolas
 - estudante;
 - * escola
 - * turma
 - disciplinas
- email
- nome de usuário
- nome legal
- password

Toda esta informação será utilizada pela aplicação para criar as relações entre contas necessárias para organizar os professores and alunos em turmas diferentes. Esta organização evita o contacto entre alunos a professores acidental out indesejado (ex: um aluno enviar um trabalho a um professor com quem não tem aulas).

3.1.2 User login

Depois de estar registado, um utilizador pode realizar o login a qualquer momento que deseja. Para realizar o login o utilizador necessita de utilizar dois bocados de informação:

- username
- password

Esta informação será enviada ao servidor para ser validada contra todas as contas encontradas dentro do sistema. A informação destas contas não é guardada em pleno texto, utilizando o método de Hash and Salt garantindo segurança para os utilizadores.

Function	Esta função garante que apenas utilizadores autorizados podem aceder ao sistema, verificando a sua identidade através de nome de utilizador e palavra-passe.
Inputs	Nome de Utilizador: Fornecido pelo utilizador. Palavra-passe: Fornecida pelo utilizador.
Source	Utilizador a tentar iniciar sessão.
Outputs	Autenticação bem-sucedida ou mensagem de erro.
Destination	Painel de controlo do utilizador ou página de erro.
Action	O sistema verifica o nome de utilizador e palavra-passe introduzidos em relação às credenciais de utilizador armazenadas. Se as credenciais forem válidas, o sistema concede acesso à conta do utilizador. Se as credenciais forem inválidas, o sistema apresenta uma mensagem de erro.
Requirements	Os dados de registo do utilizador (nome de utilizador e palavra-passe) devem ser armazenados de forma segura (Hash and Salt) no sistema.
Pre-Condition	O utilizador tem uma conta registada no sistema. O utilizador pretende iniciar sessão.
Post-Condition	Se a autenticação for bem-sucedida, o utilizador obtém acesso à sua conta e pode realizar ações autorizadas. Se a autenticação falhar, o utilizador é notificado do erro e não obtém acesso ao sistema.

3.2 Submission of work

Os professores devem ser capazes de receber submissões eletrônicas de trabalhos dos estudantes. Os estudantes devem ter a capacidade de submeter trabalhos através do sistema. Os trabalhos devem suportar formatos de documentos comuns (por exemplo, PDF, Word).

Para garantir que os trabalhos não são modificados, todos os documentos enviados pela plataforma, serão verificados pela utilização de Hash Codes. Cada ficheiro tem o seu próprio hash, sendo este modificado se qualquer tipo de modificação acontecer ao ficheiro.

Function	Esta função permite que tanto os professores como os estudantes submetam trabalhos eletronicamente através do sistema. Garante que os trabalhos submetidos suportam formatos de documento comuns, como PDF e Word.
Inputs	Ficheiro do trabalho submetido (PDF, Word ou formato suportado): Fornecido pelo utilizador (professor ou estudante).
Source	Utilizador (professor ou estudante) a iniciar a submissão do trabalho.
Outputs	Confirmação de submissão de trabalho bem-sucedida ou uma mensagem de erro.
Destination	Mensagem de confirmação ou mensagem de erro dentro do sistema.
Action	O sistema aceita e processa o ficheiro do trabalho submetido. Verifica se o formato do ficheiro é suportado (por exemplo, PDF, Word). Se o formato for suportado, o sistema confirma a submissão bem-sucedida. Se o formato não for suportado ou houver algum problema com a submissão, o sistema apresenta uma mensagem de erro.
Requirements	O sistema deve ter a capacidade de receber e processar submissões eletrônicas de documentos. Os formatos de documento suportados (por exemplo, PDF, Word) devem ser definidos e implementados.
Pre-Condition	O utilizador (professor ou estudante) tem uma conta ativa e está autenticado no sistema. O utilizador pretende submeter um trabalho.
Post-Condition	Se a submissão for bem-sucedida e o formato do documento for suportado, o sistema confirma a submissão bem-sucedida. Se houver problemas com a submissão (por exemplo, formato não suportado ou falha no carregamento), o sistema apresenta uma mensagem de erro, e a submissão não é processada.

3.3 Storing of work

Todos os ficheiros enviados pela aplicação serão guardados apenas durante a sua utilidade para avaliação e entrega da mesma. Cada ficheiro terá um código hash único e será encriptado para assegurar segurança dos trabalhos.

Cada trabalho será guardado assim que o aluno envie o mesmo para um professor. Antes do trabalho entrar no sistema, será criado um código hash com os seu estado atual e será encriptado, sendo apenas possível abrir o ficheiro depois de ser feito o login como o professor para que o documento foi enviado.

Depois de ser realizada a avaliação, a mesma será guardada separadamente do trabalho avaliado. Tendo apenas capacidade de ser visualizada se for aberta em conjunto com o trabalho correspondente pelo aluno correspondente.

Depois de todos estes passos serem executados, o documento será eliminado, deixando apenas a avaliação para ser utilizada como os professores desejarem.

3.3.1 Online storing

Os trabalhos irão ser guardados numa base de dados central. Todos os binários enviados serão guardados de acordo com os pontos feitos anteriormente. Sabendo que a maior parte dos ficheiros apenas são enviados e abertos uma vez, não é necessário uma grande solução de caching para esta aplicação.

Uma base de dados relacional consegue preencher todos os casos de utilização possíveis da aplicação. Consequentemente, a nossa solução de *storage* será extremamente simples, assim evitando várias complicações causadas por sistemas em que temos de partilhar informação entre várias bases de dados diferentes.

3.3.2 Offline storing and backups

Todos os trabalhos recebidos e enviados pelos utilizadores serão guardados nas suas respectivas máquinas. Isto é para assegurar que nenhum trabalho seja perdido ao criar três cópias do mesmo documento em três sistemas distintos.

O aluno irá ter o seu trabalho guardado na própria máquina, o nosso servidor irá ter uma cópia e o professor irá ter uma cópia criada após o download do trabalho para a avaliação. Este sistema também se aplica a avaliações. Sendo criadas cópias pela mesma lógica nos nossos servidores, na máquina do professor e na máquina do aluno.

Ao criar esta redundância de dados, é drasticamente reduzida a possibilidade da perda de dados ao interagir com o sistema, pois é virtualmente impossível todos estes pontos falharem em simultâneo.

3.4 Evaluation of Work

Os professores devem ter a capacidade de avaliar e classificar trabalhos. Os critérios de avaliação (por exemplo, rubricas) devem ser personalizáveis. Os professores devem poder fornecer comentários e atribuir pontuações a cada trabalho. O sistema deve calcular e armazenar as pontuações globais dos trabalhos.

Function	Esta função permite aos professores avaliar e classificar trabalhos submetidos pelos estudantes. Permite critérios de avaliação personalizáveis (por exemplo, rubricas) e oferece aos professores a capacidade de fornecer comentários e pontuações para cada trabalho. O sistema também calcula e armazena as pontuações globais dos trabalhos.
Inputs	Trabalho submetido (pelos estudantes). Critérios de avaliação (personalizáveis, por exemplo, rubricas) fornecidos pelo professor. Comentários e pontuações fornecidos pelo professor.
Source	Professores a aceder aos trabalhos submetidos pelos estudantes.
Outputs	Pontuações de avaliação e feedback para cada trabalho. Pontuações globais para cada trabalho.
Destination	Registo do estudante no sistema (para feedback individual). Armazenamento/base de dados do sistema (para pontuações globais).
Action	Os professores veem os trabalhos submetidos. Utilizam critérios de avaliação personalizáveis (por exemplo, rubricas) para avaliar os trabalhos. Os professores fornecem comentários e atribuem pontuações a cada trabalho. O sistema calcula as pontuações globais com base nos critérios fornecidos. As pontuações e feedback são armazenados para cada trabalho.
Requirements	Critérios de avaliação personalizáveis (por exemplo, rubricas) devem ser definidos e disponíveis no sistema. O sistema deve suportar o armazenamento de dados de avaliação.
Pre-Conditions	Os professores têm acesso aos trabalhos submetidos. Os critérios de avaliação estão estabelecidos.
Post-Conditions	Os trabalhos individuais têm associadas pontuações de avaliação e feedback. As pontuações globais para cada trabalho são calculadas e armazenadas. Os estudantes podem aceder ao seu feedback individual. Os professores podem gerir e rever as avaliações para cada trabalho.

Para este tipo de avaliação ser possível, será apresentada uma interface gráfica ao professor. Esta interface será mapeada de acordo com o documento que está a ser avaliado linha por linha. Ao ser inserido um comentário pelo professor, uma cotação ou uma correção, será guardada a posição da mesma, sendo possível assim, reconstruir a correção do professor quando esta for ser vista pelo aluno. Isto evita a necessidade de modificar os documentos enviados pelos alunos em si.

3.5 Notifications

O sistema deve notificar os estudantes automaticamente quando os seus trabalhos forem avaliados. As notificações devem incluir os resultados da avaliação e os comentários.

As notificações serão simples notificações de IOS/Android/Windows/MacOS/Linux, utilizando os sistemas de notificações oferecidos por estas plataformas.

Function	Esta função envolve a notificação automática dos estudantes quando os seus trabalhos forem avaliados. As notificações devem incluir os resultados da avaliação e o feedback.
Inputs	Resultados da avaliação e feedback gerados durante a avaliação do trabalho.
Source	O sistema, especificamente o processo de avaliação de trabalhos.
Outputs	Mensagens de notificação enviadas aos estudantes correspondentes.
Destination	Canais de comunicação dos estudantes (por exemplo, email, notificações no sistema).
Action	Após a conclusão da avaliação dos trabalhos, o sistema gera mensagens de notificação. Estas mensagens incluem os resultados da avaliação (pontuações) e o feedback. O sistema envia estas notificações aos respetivos estudantes.
Requirements	Um mecanismo para a geração de notificações automáticas. Acesso às informações de contacto dos estudantes (por exemplo, endereços de email).
Pre-Conditions	As avaliações dos trabalhos foram concluídas pelos professores. As informações de contacto dos estudantes estão disponíveis e atualizadas.
Post-Conditions	Os estudantes recebem notificações automáticas contendo os resultados da avaliação dos seus trabalhos e o feedback. Os estudantes podem aceder aos seus resultados de avaliação e ao feedback através dos seus canais de comunicação escolhidos (por exemplo, email, notificações no sistema).

3.5.1 Notifications for students

Cada notificação para estudantes será personalizada de acordo com o trabalho que representa, disciplina e professor a quem foi enviado. Uma notificação poderá conter mensagens como os seguintes exemplos:

- Trabalho enviado a professor:
 - O trabalho x foi enviado ao professor y — 19:04.
- Trabalho recebido por professor:
 - O trabalho x foi recebido pelo professor y — 19:06.
- Avaliação recebida por aluno:
 - Recebeu uma avaliação nova do trabalho x — 19:07

3.5.2 Notifications for professors

Tal como para os estudantes, a informação apresentada nas notificações dos professores irá variar de acordo com os trabalho recebido e condições como o tempo a que as mesmas foram enviadas:

- Novo trabalho para avaliar recebido:
 - Trabalho **novo** recebido de aluno x da turma y da disciplina z . — 19:10
- Avaliação recebida:
 - A **avaliação** do trabalho x foi recebida pelo aluno y da turma z da disciplina h — 19:16

Para além das notificações explicadas até agora, a aplicação não deve enviar qualquer outro tipo de notificação aos seus utilizadores. Isto é para evitar o envio excessivo de notificações que acontece na indústria.

4 Non-Functional Requirements

- Performance
- Security
- Usability
- Interação simples
 - interface gráfica
 - interface de linha de comandos
- Scalability
- Stability
- Easy notifications

4.1 Performance

O sistema deve ser capaz de lidar com um grande número de submissões de trabalhos e avaliações simultaneamente. Os tempos de resposta para aceder aos trabalhos e avaliações devem ser rápidos.

Os tempos de resposta serão alcançados pela implementação de um sistema de caching local. Este sistema irá fazer o download de todos os documentos recebidos ao ligar a aplicação ao invés de fazer o download apenas quando os documentos estiverem a ser abertos. Este download deve ser feito assincronamente para evitar o bloqueio da funcionalidade do resto da aplicação. Permitindo a interação com os resto das funcionalidades da aplicação em tempo real durante o download de outros ficheiros que não se encontram a ser utilizados.

4.2 Security

Os dados dos utilizadores, trabalhos e avaliações devem ser armazenados de forma segura. A autenticação de utilizadores deve utilizar métodos de encriptação seguros. Apenas utilizadores autorizados (professores e estudantes) devem ter acesso a dados introduzidos no sistema por outros utilizadores.

Os métodos utilizados para assegurar a segurança dos utilizadores e dos documentos dentro da aplicação já foram explicados anteriormente neste documento.

Para assegurar que os documentos não são modificados depois de entrarem dentro do sistema da aplicação, irá ser utilizado um sistema de hashing que identificará o estado do documento quando este entra na aplicação pela primeira vez. Qualquer mudança ao documento (ex, nome) será refletida numa mudança do seu código hash. Isto permite que a aplicação consiga assegurar a integridade de todos os documentos aos seus utilizadores, deixando estes saber que não é necessário desconfiar os documentos enviados pela aplicação.

Em termos de segurança de utilizadores e das suas contas, toda a informação de todos os utilizadores será guardada e protegida utilizando um método do tipo *Hash and Salting*, isto evita que uma falha na segurança da base de dados não signifique o comprometimento dos dados dos utilizadores.

4.3 Usability

A interface do utilizador deve ser intuitiva e de fácil utilização. Professores e estudantes devem conseguir navegar facilmente e executar tarefas dentro do sistema.

Tendo em conta que é impossível criar uma interface e *user experience* que seja adequada a todos os utilizadores, a aplicação terá duas maneiras de interação:

- Interação gráfica.
- Interação por linha de comandos.

4.3.1 Interação gráfica

A interação gráfica será a maneira principal de interagir com a aplicação e as suas funcionalidades. O design e princípios utilizados para desenvolver estas interfaces devem seguir todos os princípios de design e regras impostos pelos sistemas em que a aplicação está a ser desenvolvida, seja isto um ambiente mobile ou um ambiente desktop.

4.3.2 Interação por linha de comandos

Para nos assegurar-mos da portabilidade da aplicação a garantir a sua utilidade em sistemas menos suportados e no futuro, será criada uma interface da aplicação que será utilizada pela linha de comandos. Esta versão apenas deve utilizar tecnologias que se encontrem em sistemas que são *POSIX compliant* para garantir a sua portabilidade e funcionalidade em vários sistemas que seguem estes princípios.

Esta versão da aplicação também será utilizada para o desenvolvimento de outras interfaces para o programa. Estas interfaces não serão desenvolvidas por nós, mas sim pelas empresas que se encontram a utilizar a aplicação e necessitem de outras maneiras de utilizar a aplicação que não forem desenvolvidas por nós.

4.4 Scalability

O sistema deve ser projetado para crescer à medida que o número de utilizadores e submissões aumenta ao longo do tempo. A utilização de tecnologias provadas em mercado será uma necessidade durante a construção do sistema.

Para Isto ser assegurado, a aplicação base será apenas em linha de comandos, criando um sistema fácil de ser expandido e modificado caso seja necessário. Os servidores irão utilizar tecnologias Open Source para ser possível mudar os mesmos para as necessidades da aplicação e para ter acesso a mais suporte externo para evitar futuros problemas que normalmente acompanham plataformas proprietárias.

4.5 Stability

O sistema deve estar disponível e ser fiável, com um tempo de inatividade mínimo.

Cópia de Segurança e Recuperação de Dados:

Deve haver cópias de segurança regulares de dados (trabalhos, avaliações, contas de utilizador). Deve existir um plano para a recuperação de dados em caso de falhas no sistema.

Simplicidade:

Sendo desenvolvida com tecnologias utilizadas pela indústria e tecnologias acessíveis, é assegurada a simplicidade da aplicação e consequentemente, a simplicidade da manutenção da mesma por nós e pelo cliente. Isto deixa o cliente resolver os seus problemas sem necessitar de recursos externos que podem colocar em causa a sua utilização da aplicação.

4.6 Notification Delivery

As notificações da aplicação devem ser simples de ler de configurar. Ao utilizar as opções oferecidas pelas plataformas utilizadas, o utilizador será capaz de mudar as notificações de maneira a integrar as mesmas com a sua maneira preferível de trabalhar, sem necessitar de ter alguma experiência com programação ou com o desenvolvimento ou funcionamento da aplicação.