**Gestor de Riscos e Testes**

Nesta secção, o desenvolvimento do projeto é analisado de forma a identificar os pontos vulneráveis e refletir sobre estes, de forma a reduzir a probabilidade da sua ocorrência. Para além de antecipar os pontos vulneráveis, esta secção engloba também a análise de testes mais adequados a realizar ao software desenvolvido.

**Análise de pontos vulneráveis**

A análise dos pontos mais vulneráveis corresponde aos riscos a que o projeto está sujeito. É um conjunto de acontecimentos que desequilibram o funcionamento e desenvolvimento de um projeto. Associados ao projeto a ser desenvolvido, e adaptável às três etapas de desenvolvimento, são identificados os seguintes pontos:

1. Incumprimento da data de entrega do projeto;
2. Ao longo do desenvolvimento do projeto encontrar uma falha no planeamento que altere a estrutura inicialmente prevista para o projeto;
3. Um ou múltiplos elementos integrantes do grupo desistirem da realização do projeto;
4. Perda de informação desenvolvida, ao eliminar informação importante;
5. Tarefas dependentes em cadeia que dão origem a atrasos sucessivos;
6. Avarias nas plataformas de desenvolvimento dos elementos da equipa;
7. Mudanças/ restruturações derivadas de riscos anteriormente não identificadas.

Ao identificar a ocorrência de um risco é essencial recolher informação sobre este à medida a que esta se torna disponível. É necessário reanalisar o risco e priorizar em função da sua relevância. Os riscos identificados têm prioridades distintas ao longo do desenvolvimento do projeto. É então, indispensável, ao longo do desenvolvimento do projeto, analisar as prioridades dos riscos e reavaliar as suas consequências.

O risco 1, o incumprimento do prazo de entrega definido, é adaptável ás três etapas do projeto e ás suas respetivas datas de entrega. Pode surgir derivável de vários riscos e conjugação destes. Como tal, é necessário avaliar detalhadamente as causas e reajustar recursos de forma a reduzir os atrasos gerados. É então, um risco latente que deve ser evitado, monotonizando os restantes riscos.

O risco 2 surge ao longo da evolução do projeto e está associado a erros no planeamento do desenvolvimento deste. Pode surgir derivado de múltiplos acontecimentos, sendo identificados como mais prováveis, erros no planeamento da estrutura do software e/ou erros na estrutura do planeamento das etapas do projeto. Para evitar e reduzir a probabilidade de ocorrem erros associados ao risco 2 é indispensável um planeamento exaustivo. Este planeamento deverá considerar múltiplas abordagens, simulações do projeto e identificar possíveis incompatibilidades.

O risco 3 engloba o risco de, ao longo de todo o desenvolvimento, um ou vários elementos pertencentes à equipa do projeto renunciarem do seu cargo. Apesar das causas associadas à ocorrência do risco 3 serem maioritariamente exteriores, é essencial desenvolver um plano estratégico para colocar em prática, na sua eventualidade. Como estratégia para reduzir os efeitos do risco em questão, deve ser elaborado um plano de realocação de trabalho. Deve ser identificado o(s) elemento(s) da equipa mais adequando para substituir as funções do elemento que abandona o seu cargo, de modo a encontrar um novo equilíbrio na distribuição de tarefas.

O risco 4, a perda de informação desenvolvida, aumenta com o numero de indivíduos envolvidos na manipulação de documentos importantes. Para evitar a perda de informação, todos os membros da equipa, devem saber como manipular corretamente as plataformas utilizadas. E toda a informação deve ser salvaguardada em múltiplas plataformas, incluindo os computadores individuais dos membros da equipa.

O risco 5 está associado a atrasos no desenvolvimento. No planeamento das diferentes tarefas a desenvolver, é inevitável a existência de dependências entre tarefas. O que, ao ocorrer um atraso numa das tarefas a ser desenvolvida, derivará num consequente atraso nas tarefas dependentes. Para quebrar a cadeia de atrasos deverá, mais uma vez, ser implementado um plano de redistribuição de funções para reduzir o seu impacto.

O risco 6, a avaria de plataformas utilizadas no desenvolvimento do projeto, envolve fatores fora do controlo dos membros da equipa. No entanto, devem ser planeados mecanismos de reação no caso da sua ocorrência. Estes devem envolver, mais uma vez, realocação de funções entre os membros da equipa de forma a que o impacto seja reduzido.

O risco 7 envolve riscos não previsto no planeamento. Na eventualidade de surgir um risco nesta categoria este deverá ser estudado e avaliado de modo a obter um plano de reação. O plano deverá ter como objetivo final reestabelecer o bom funcionamento do projeto.

**Análise de testes a desenvolver**

A análise de testes tem como propósito verificar se o programa desenvolvido realiza as operações requeridas e identificar defeitos para que estes sejam corrigidos. É importante realçar que os testes efetuados têm o objetivo de identificar possíveis erros, e não demonstrar a ausência destes. Para tal os seguintes testes devem ser desenvolvidos:

1. Analisar o código fonte;
2. Testar todas as funcionalidades pretendidas;
3. Desenvolver inputs que podem originar problemas no programa;
4. Analisar o design.

Antes de começar a avaliar o programa através dos testes definidos, durante o desenvolvimento do software, duas questões devem ficar verificadas. Se o programa a ser desenvolvido coincide com o estabelecido no enunciado do projeto. E se o produto está a ser desenvolvido corretamente e eficientemente. De seguida os restantes testes devem ser realizados.

O teste 1, que engloba a análise do código fonte, deve ser efetuado anteriormente a todos os restantes testes. É essencial ter conhecimento de todo o código desenvolvido e de todo o seu funcionamento.

O teste 2 tem como objetivo testar as funcionalidades do programa e verificar se estas estão todas operacionais. É imprescindível realizar, no mínimo, um teste redirecionado para cada uma das funcionalidades.

O teste 3 pretende, através de combinações de diferentes inputs, não esperados numa utilização correta do programa, tornar o sistema mais robusto.

O teste 4 é uma avaliação do design verificando o seu bom funcionamento e compatibilidade com o requerido.

De seguida é apresentado uma adaptação dos testes a realizar a cada uma das etapas:

**Etapa 1**

A etapa 1 requer as funcionalidades F1 e F2, que são, respetivamente, a comunicação por sockets entre as aplicações do sistema e a gestão de credenciais para a utilização das aplicações.

Todos os testes mencionados (1-4) devem ser realizados. Os testes 1 e 4 não têm especificações distintas para as diferentes funcionalidades. Aplicando diretamente às funcionalidades F1 e F2 deve ser realizado o seguinte:

**Tarefa F1:**

Para o teste 2 a comunicação entre as aplicações tem de ser verificada. Para tal a Aplicação para o Profissional de Saúde (APS) deve enviar informação para a Aplicação Central (AC). De seguida, através da aplicação AAS, essa informação deve ser lida.

Por exemplo, enviar “estou a enviar dados – app APS”, e os dados enviados devem estar disponíveis na base de dados de registo da aplicação central. Para posteriormente estarem disponível para leitura através da aplicação AAS.

No teste 3 devem ser incluídos testes com caracteres diferentes e mensagens de tamanhos distintos.

Por exemplo, “12345 projetoRC 1etapa” e “de que tamanho pode ser a mensagem? Tem limite?”.

**Tarefa F2:**

Para o teste 2 devem ser introduzidas, na base de dados, autenticações para todas as aplicações. De seguida essas autenticações devem ser introduzidas nas respetivas aplicações. Devem ser também introduzidas, nas aplicações, testes de autenticações incorretas, que não devem ter uma autenticação bem-sucedida.

Por exemplo, introduzir na base de dados as autenticações seguintes: “profissional\_de\_saude@gmail.com” e “sou\_saude”, “profissional\_de\_segurança@gmail.com” e “sou\_segurança”, e “gestor\_sistema@gmail.com” e “sou\_gestor”, como email e palavra-passe de autenticação das aplicações APS, AAS e AGS, respetivamente. Posteriormente, o funcionamento das respetivas autenticações deve ser testado. Tendo em consideração que apenas as autenticações introduzidas devem ser autorizadas a continuar a navegar na aplicação.

Para o teste 3, mais testes que não devem ser autenticados devem ser testados. Por exemplo, trocar algumas letras das autenticações corretas e verificar a recusa dessas modificações quando realizar o login nas aplicações.

**Etapa 2**

A etapa 1 requer as funcionalidades F3, F4, F5, F6 e F7, que são, respetivamente, a gestão dos seus registos por parte do profissional de saúde e agente de segurança, gestão das contas registadas nas aplicações por parte do administrador do sistema, o registo de crimes por parte do profissional de saúde e a consulta dos crimes registados, com ou sem filtros, por parte dos agentes de segurança.

Todos os testes mencionados (1-4) devem ser realizados. Os testes 1 e 4 não têm especificações distintas para as diferentes funcionalidades. Aplicando diretamente às funcionalidades da etapa 2 deve ser realizado o seguinte:

**Tarefa F3 e F4:**

Para o teste 2 deve ser criado um perfil para um profissional de saúde. E verificar que foi criado como o suposto. De seguida alterar-se esse perfil e verificar a mudança. Posteriormente eliminar-se o perfil e, mais vez, verificar as modificações.

Para o teste 3, com um email já utilizado num registo tentar criar uma nova conta, que deverá produzir uma mensagem de erro. Ao alterar os dados de uma conta uma mensagem de erro também deverá ser apresentada.

**Tarefa F5:**

Para testar as funções da AGS, esta deve conseguir validar ou eliminar o registo de um profissional de saúde e dos agentes de segurança. O seguinte teste deve ser implementado. Iniciar APS e efetuar um registo de uma nova conta. Com a AGS deverá ser possível visualizar o pedido de validação de registo efetuado pelo profissional de saúde e este deverá ser aceite. Com a APS, depois do registo ser aceite, deve ser possível efetuar o login com o registo em questão. O mesmo procedimento deve ser efetuado para a AAS.

Esta funcionalidade não requer inputs não standards.

**Tarefa F6:**

Para testar as funcionalidades de F6, ao entrar na aplicação APS, depois da autenticação, ao registar na plataforma uma ocorrência de um crime este deve ser registado pela AC na base de dados.

Para testar a resistência a inputs os diferentes campos necessitam de testes distintos. O campo da data deverá aceitar apenas datas válidas, por exemplo, deverá ser recusado o registo com uma data futura. O campo da hora deverá apenas ser validado com valores entre de 0 a 23 na hora e de 0 a 59 nos minutos. O email inserido apenas deverá ser aceite se fizer parte dos registes validados pela AGS.

**Tarefa F7:**

Para testar as funcionalidades de F7, ao efetuar o registo na aplicação AAS, depois do processo de autenticação, todos os registos da base de dados devem ser visualizados. Ao aplicar filtros apenas os dados do respetivo filtro devem ser visualizados.

Esta funcionalidade não requer inputs não standards.

**Etapa 3**

A etapa 3 requer as funcionalidades F8, F9, F10, F11 e F12, que são, respetivamente, a possibilidade de o profissional de saúde efetuar um registo anónimo, a funcionalidade do botão de alarme, o sistema de apoio às funcionalidades das aplicações, o chat e todas as funcionalidades adicionais.

Todos os testes mencionados (1-4) devem ser realizados. Os testes 1 e 4 não têm especificações distintas para as diferentes funcionalidades. Aplicando diretamente às funcionalidades da etapa 3 deve ser realizado o seguinte:

**Tarefa F8:**

Para testar o funcionamento de um registo anónimo deve ser efetuado uma simulação de um registo com o campo do email em anónimo, através da APS. De seguida consultar a informação do registo e verificar a ausência de uma associação entre o registo do crime e o profissional de saúde. Para tal, ao consultar o crime na aplicação AAS, o email do profissional do registo deverá aparecer como o email anónimo.

Esta funcionalidade não requer inputs não standards.

**Tarefa F9:**

Para verificar o funcionamento do botão de alarme na APS deve ser simulada uma situação de teste. Para tal, ao aceder à APS deve ser acionado o protocolo do botão de alarme, acionando o mesmo. De seguida, com a AAS devidamente iniciada, uma notificação consequente ao acionar o “botão de alarme” deve ser apresentada no ecrã.

Casos como o acionamento do botão indevidamente devem ser testado de modo semelhante ao descrito no paragrafo anterior. No entanto, o objetivo será obter notificações distintas.

**Tarefa F10:**

Com o objetivo de testar o sistema de apoio o teste seguinte deve ser aplicado a todas as aplicações envolvidas. Depois de realizada a autenticação, ao introduzir o comando “help”, uma mensagem de ajuda, com a explicação das respetivas aplicações, deverá aparecer no ecrã.

Para testar múltiplos inputs, palavras semelhantes a “help”, devem ser introduzidas. Como “jelp”, “helç” ou “hekp”. Todas estas devem gerar mensagens de erro.

**Tarefa F11:**

Para testar o chat entre a vitima e o agente de segurança, ambas as aplicações (APS e AAS) devem ser devidamente iniciadas. E ao enviar mensagens de uma das aplicações a outra deverá receber.

**Tarefa F12:**

Todas as funcionalidades dentro da categoria F12 devem ser testadas e apresentar comportamentos previstos. A tolerância a inputs deverá ser testada, em semelhança das tarefas anteriores.