Risk-Based Testing Approach for the IMGU Application

Table of Contents

[Understanding Risks in Software Testing 1](#_Toc170486275)

[IMGU Application: Functionality and Domains 1](#_Toc170486276)

[Risk Assessment 3](#_Toc170486277)

[Analysis and Management of Specific Potential Risks in Various Domains 4](#_Toc170486278)

[Implementation of Testing Methods 7](#_Toc170486279)

## Understanding Risks in Software Testing

Risk-based testing (RBT) is a strategy in software testing where testing resources are allocated based on the risk levels of various functionalities and components of the application. This approach prioritizes testing activities to focus on the areas with the highest risk of failure and the greatest potential impact on the application.

Based on the definition, a approach to risk-based testing is desighned.

Action Plan for Creating a Testing Strategy for IMGU Application

1. Functionality Description
2. Risk Level Assessment Based on Functionality
3. Analysis and Management of Specific Potential Risks in Various Domains
4. Implementation of Testing Methods

## IMGU Application: Functionality and Domains

IMGU (Initial Management fo Genital Ulcer) designed for healthcare professionals to assist in prescribing empirical treatment for symptoms of genital ulcers.

**Current Application Features:**

* **Decision Tree Interface:** Uses structured questions to gather patient information and determine the best diagnostic and therapeutic steps.
* **Decision-Making Algorithms:** Built-in algorithms based on up-to-date medical guidelines and protocols to aid in informed decision-making.
* **Diagnostic Test Recommendations:** Provides suggestions for necessary diagnostic tests.
* **Empirical Treatment Recommendations:** Offers recommendations for empirical treatment.

**Potential Features for Real-World Implementation:**

* **Patient History Collection and Recording**
* **Physical Examination Data Entry**
* **Diagnostic Results Entry**
* **Patient Visit Documentation and Reporting**
* **Security and Confidentiality:** Data storage and encryption.
* **User Support**
* **Free and Paid Version**

The potential and existing features of the application can be correlated with the areas of risk.

|  |  |  |
| --- | --- | --- |
| Functionality | Primary Risk Domains | Level of Layer for Testing |
| **Patient History Collection and Recording** | **UX/UI Risks, Regulation Risks** | L1-L4 |
| **Physical Examination Data Entry** | **UX/UI Risks, Regulation Risks** | L1-L4 |
| **Diagnostic Results Entry** | **UX/UI Risks, Regulation Risks** | L1-L4 |
| **Patient Visit Documentation and Reporting** | **UX/UI Risks, Regulation Risks** | L1-L4 |
| **Decision Tree Interface** | **Technical Risks,** **Clinical Risks, Regulation Risks** | L1-L4 |
| **Decision-Making Algorithms** | **Clinical Risks** | L1-L3 |
| **Diagnostic Test Recommendations** | **Clinical Risks** | L1-L4 |
| **Empirical Treatment Recommendations** | **Clinical Risks** | L1-L4 |
| **Security and Confidentiality** | **Security Risks** | L4 |
| **User Support** | **Exploitation Support Risks** | L4 |
| **Free and Paid Version** | **Business Risks** |  |

## Risk Assessment

In conducting risk-based testing, it is essential to assess the risk levels associated with various types of errors. This assessment employs a base risk matrix, calculating risk levels using the formula: Likelihood x Consequence = Level of Risk. To formulate a risk management strategy, a risk strategy matrix is utilized. This approach indicates that areas such as Clinical Risks, Technical Risks, UX/UI Risks, and Regulation Risks necessitate a "Reduce the Impact/Reduce the Likelihood" strategy. Errors in these domains pose critical threats to the functionality of the medical application and thus warrant maximum attention during the development of the testing strategy. For security-related risks, it is recommended to transfer these risks by either securing insurance or employing a platform that ensures the application's security. Risks associated with Business Risks are accepted due to their relatively low impact.

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Автоматически созданное описание

Figure 1 Risk Matrix and Risk mitigation Strategy (Wijnia, Ype. 2012).

**Significant Risks in the IMGU Application**

1. **Clinical Risks:** Including incorrect diagnosis and treatment errors, which have medium likelihood and high consequence.
2. **Technical Risks:** Such as system crashes and performance issues, which pose substantial risks due to their potential to disrupt application functionality.
3. **UX/UI Risks:** Involving user interface flaws and user experience problems, which can significantly affect usability and user satisfaction.
4. **Security Risks:** Despite being less likely, the potential for data breaches and unauthorized access remains high due to the severe consequences.
5. **Regulation Risks:** Non-compliance with regulatory requirements, which can lead to legal issues and loss of user trust.

**Lower-risk areas include:**

* **Exploitation Support Risks:** Although frequent, these risks have minimal impact.
* **Business Risks:** Assessed as low due to their minor impact on the overall application functionality.

Based on this analysis, a comprehensive risk management strategy has been developed:

* **Clinical Risks, Technical Risks, UX/UI Risks, and Regulation Risks:** Adopt a "Reduce the Impact/Reduce the Likelihood" strategy. Given their potential to cause critical problems, these risks will receive the highest priority in testing and mitigation efforts.
* **Exploitation Support Risks:** Aim to reduce the frequency of potential problems in this area.
* **Security Risks:** Due to their high impact despite low likelihood, these risks will be transferred through insurance or by using platforms that guarantee application security.
* **Business Risks:** These will be accepted as their impact is relatively low.

This detailed analysis and targeted strategy ensure that all significant risks are appropriately managed, thereby enhancing the reliability and safety of the IMGU application.

## Analysis and Management of Specific Potential Risks in Various Domains

This chapter evaluates the risks of the existing application, rather than its potential implementation.

**1. Clinical Risks**

**Risk Types:**

1. **Incorrect Empirical Treatment Recommendation**
   * **Description:** The application may suggest incorrect empirical treatments, potentially leading to patient deterioration, antibiotic resistance, or other severe consequences.
   * **Likelihood:** High, if algorithms are not precise or are based on outdated or incorrect data.
   * **Impact:** High, as incorrect treatment can seriously harm the patient.
   * **Priority:** High
2. **Incorrect Additional Tests Recommendation**
   * **Description:** The application may recommend unnecessary or incorrect tests, leading to unnecessary expenses and diagnostic delays.
   * **Likelihood:** High, if algorithms do not account for all clinical nuances.
   * **Impact:** Medium, primarily affecting cost and diagnostic speed rather than directly impacting patient health.
   * **Priority:** Medium
3. **Incorrect Diagnosis**
   * **Description:** The application may suggest incorrect or incomplete diagnoses, potentially leading to patient deterioration, antibiotic resistance, or other severe consequences.
   * **Likelihood:** High, if algorithms are not precise or are based on outdated data.
   * **Impact:** High, as incorrect treatment can seriously harm the patient.
   * **Priority:** High

**Risk Management:**

* Continuous updates of clinical data and algorithms.
* Multi-level review of recommendations by physicians.
* Implementation of alert systems for potential contraindications and side effects.
* Regular audit and optimization of test recommendation algorithms.
* Inclusion of test necessity evaluation based on clinical context.
* User feedback integration to improve algorithms.

**Testing Methods:**

* **Manual Testing:** Physicians and specialists verify the accuracy of recommendations based on clinical cases. Creation of test cases based on real clinical scenarios and evaluation by medical experts.
* **Expert Involvement Testing:** Organize testing sessions involving medical experts to evaluate algorithm accuracy and recommendations. Regular review sessions with physicians to check and improve algorithms.
* **User stories Testing.** Testing is based on clinical cases or special stories relevant to the App use.

**2. Technical Risks**

**Risk Types:**

1. **Algorithm Coding Errors / Incorrect Transfer of Medical Guidelines**
   * **Description:** Incorrect translation of guidelines into machine-readable code may lead to inaccurate diagnoses.
   * **Likelihood:** High, if the application does not undergo sufficient testing.
   * **Impact:** High, as this directly affects recommendation accuracy.
   * **Priority:** High
2. **Application Code Errors**
   * **Description:** Technical errors in the algorithms may result in incorrect recommendations.
   * **Likelihood:** High, if the application does not undergo sufficient testing.
   * **Impact:** High, as this directly affects recommendation accuracy.
   * **Priority:** High

**Risk Management:**

* Intensive testing of all algorithms before and after deployment.
* Regular code and logic audits.
* Implementation of quality control and algorithm certification systems.
* Implement rigorous procedures for transferring medical guidelines into the application, including thorough review and validation by medical experts.

**Testing Methods:**

* **Unit Testing:** Verification of individual components and functions to ensure they work correctly.
* **Integration Testing:** Checking the interaction of all parts of the algorithm and their proper functioning within the overall system.
* **End-to-End Testing:** Comprehensive testing from data input to result generation to verify all stages of algorithm operation.
* **Code Review:** Regular code inspections involving multiple developers to detect errors and improve code quality.

**3. UX/UI Risks**

**Risk Types:**

1. **Inconsistent Design**
   * **Description:** Inconsistent design elements (such as varying button styles, fonts, and colors) can confuse users and reduce the overall usability of the application.
   * **Likelihood:** High, if the interface is not intuitive or informative.
   * **Impact:** High, as users may stop using the application.
   * **Priority:** High
2. **User Errors**
   * **Description:** Physicians may misinterpret data or recommendations from the application.
   * **Likelihood:** High, if the interface is not intuitive or lacks sufficient support materials.
   * **Impact:** Medium, as errors can be identified and corrected during data interpretation.
   * **Priority:** High
3. **Overload of Information**
   * **Description:** Presenting too much information at once can overwhelm users, making it difficult for them to focus on and understand the most critical information.
   * **Likelihood:** High, if the application does not properly organize and prioritize information.
   * **Impact:** High, as this can lead to user frustration and incorrect usage.
   * **Priority:** High

**Risk Management:**

* Development of an intuitive interface and detailed documentation.
* Information Prioritization: Organize and prioritize information to avoid overwhelming users. Present information in a clear and manageable way.
* Regular training and user education.
* Implementation of verification and confirmation systems for user actions.

**Testing Methods:**

* **Usability Testing:** Involving real users to perform typical tasks and observe their interactions with the application.
* **A/B Testing:** Comparing different interface versions to determine the best option.
* **Focus Group Testing:** Conducting sessions with multiple users to discuss and evaluate the interface.
* **Cognitive Walkthrough:** Step through the application as if you were a new user, identifying potential points of confusion or difficulty.

**4. Regulation Risks**

**Risk Types:**

1. **Non-compliance with Regulatory Requirements**
   * **Description:** The application may not comply with regulations such as HIPAA, GDPR, or local medical standards, leading to legal issues and fines.
   * **Likelihood:** High, if the application does not undergo sufficient compliance checks.
   * **Impact:** High, as non-compliance can result in severe legal and financial consequences.
   * **Priority:** High

**Risk Management:**

* Continuous auditing for regulatory compliance.
* Regular internal and external compliance audits.
* Creation of compliance checklists to verify all regulatory aspects.
* Regular knowledge updates through training and seminars for the development and testing teams on current regulatory requirements.

**Testing Methods:**

* **Compliance Audits:** Regular internal and external audits to verify regulatory compliance.
* **Compliance Checklists:** Creation of checklists to verify all aspects related to regulatory requirements.
* **Knowledge Updates:** Conducting training and seminars for the development and testing teams on current regulatory requirements.

By addressing these risks and implementing thorough testing and risk management strategies, the reliability and safety of the IMGU application can be significantly improved.

## Implementation of Testing Methods

To implement the Testing Strategy on a first stage the following test where perfomed:

1. To avoid Incorrect Transfer of Medical Guidelines – testing branches of Apps desicion logic. Code reweiw and integration testing(the results are available in Github)
2. To avoid Algoruthm Code Errors were perfomed End-to End testing of the ODK file. (the results are available in Github)
3. To avoid Incorrect Empirical Treatment Recommendation were perfomed testing based on Usr Stories. (the results are available in Github)

## Results

Based on the test results, it was recommended to modify the application's algorithm to correct the identified errors and reduce the level of risks.

**References**

Wijnia, Ype. (2012). Asset Risk Management: Issues in the Design and Use of the Risk Matrix. 10.1007/978-0-85729-493-7\_81.