|  |
| --- |
|  |
| BTWBeheer  Research reports – Individual project |
| |  |  |  | | --- | --- | --- | | Nies,Midas M.F.A. | Demand based - Semester 3 | Software engineering HBO ICT Fontys | |

Contents

[About this document 1](#_Toc156220457)

[Research topic 1: Security 2](#_Toc156220458)

[How can I make my application as secure as possible? 2](#_Toc156220459)

[What security measures are important to consider when making an enterprise application? 2](#_Toc156220460)

[What security measures are applicable to my project and development stack, and why? 3](#_Toc156220461)

[How will I apply these security measures? 4](#_Toc156220462)

[Research topic 2: Bookkeeping 7](#_Toc156220463)

[What are the most important functional requirements for the development of an effective and user-friendly web application for bookkeeping, made for small companies. 7](#_Toc156220464)

[What are the most important things that Dutch small companies need to bookkeep? 7](#_Toc156220465)

[What bookkeeping programs already exist, and what are their limitations? 7](#_Toc156220466)

[What are the essential functionalities which Dutch small companies need? 7](#_Toc156220467)

[Bibliography 7](#_Toc156220468)

# About this document

In this document I will keep track of my two required research topics for this semesters project, about a bookkeeping program. I will discuss my two research topics which both will have multiple sub questions. The main research topic must therefore be a broad but project specific topic.

# Research topic 1: Security

This research topic must and will be about security, as required by the teachers. My project is a bookkeeping program, which will probably contain important and secret documents from companies around the country. So, keeping this information private and secure should be an important part of the project. I will therefore look into secure password hashing, 2 factor authentication, and secure web tokens to use the API.

## How can I make my application as secure as possible?

### What security measures are important to consider when making an enterprise application?

The following is a summary of an expert interview with Kevin Norbert a friend of mine which works at an ICT company specialized in enterprise application security.

“””

To enhance the security of your bookkeeping program developed on a specific stack, it's crucial to explore security measures tailored to the technologies you're utilizing. Consider the following aspects based on common components of a typical web application stack:

**Programming Language and Framework:**

* Investigate security features provided by your chosen programming language and framework. For example, if you're using Python with Django or Flask, understand their security mechanisms.

**Database Security:**

* Implement secure coding practices for database interactions.
* Utilize parameterized queries or prepared statements to prevent SQL injection attacks.
* Set strong authentication mechanisms for database access.

**Web Server Security:**

* Configure your web server (e.g., Apache, Nginx) to follow best security practices.
* Use HTTPS to encrypt data in transit, and ensure the server is properly configured to handle secure connections.

**Authentication Libraries:**

* If using third-party authentication libraries, ensure they follow industry standards and best practices.
* Regularly check for updates and security patches for these libraries.

**API Security:**

* If your application involves APIs, implement secure authentication methods like OAuth 2.0 or API tokens.
* Validate and sanitize input data to prevent injection attacks.

**Frontend Security:**

* Employ secure coding practices for your frontend technologies (e.g., React, Angular).
* Protect against Cross-Site Scripting (XSS) and Cross-Site Request Forgery (CSRF) attacks.

**Dependency Scanning:**

* Regularly scan and update third-party dependencies to patch any known vulnerabilities.
* Use tools like OWASP Dependency-Check to identify and address security issues in your project dependencies.

**Container Security (if using containers):**

* If utilizing containerization (e.g., Docker), implement container security best practices.
* Regularly scan container images for vulnerabilities.

**Cloud Service Security (if applicable):**

* If deploying your application in the cloud, follow security best practices for the chosen cloud provider (e.g., AWS, Azure, Google Cloud).
* Utilize identity and access management (IAM) to control access to cloud resources.

**Logging and Monitoring:**

* Set up comprehensive logging and monitoring to detect and respond to security incidents promptly.
* Implement alerting for suspicious activities or potential security breaches.

By delving into security features and considerations specific to your development stack, you can tailor your security strategy to address potential vulnerabilities and strengthen your bookkeeping program against various threats.

“””  
(Norbert, 2023)

### What security measures are applicable to my project and development stack, and why?

After the interview I have looked into what things would have a positive impact on my application, the following are the thing I have decided are applicable for my project.

**Database Security:**

* Implement secure coding practices for database interactions.
* Utilize parameterized queries or prepared statements to prevent SQL injection attacks.
* Set strong authentication mechanisms for database access.

**Web Server Security:**

* Use HTTPS to encrypt data in transit, and ensure the server is properly configured to handle secure connections.

**API Security:**

* If your application involves APIs, implement secure authentication methods like OAuth 2.0 or API tokens.

**Container Security:**

* If utilizing containerization (e.g., Docker), implement container security best practices.
* Regularly scan container images for vulnerabilities.

**Cloud Service Security (if applicable):**

* If deploying your application in the cloud, follow security best practices for the chosen cloud provider (e.g., AWS, Azure, Google Cloud).
* Utilize identity and access management (IAM) to control access to cloud resources.

**Logging and Monitoring:**

* Set up comprehensive logging and monitoring to detect and respond to security incidents promptly.
* Implement alerting for suspicious activities or potential security breaches.
* By delving into security features and considerations specific to your development stack, you can tailor your security strategy to address potential vulnerabilities and strengthen your bookkeeping program against various threats.

### How will I apply these security measures?

**Database Security:**

* Implement secure coding practices for database interactions.
  + I will be using a common practice for express applications. (Krebs, 2021)
* Utilize parameterized queries or prepared statements to prevent SQL injection attacks.
  + Database queries will be generated by Sequilize which is considered a safe modeling tool. (Ali, 2023)
* Set strong authentication mechanisms for database access.
  + The database credentials are reset every month and contains a random username and password made up of 32 random characters, which is considered a really safe login.

**Web Server Security:**

* Use HTTPS to encrypt data in transit, and ensure the server is properly configured to handle secure connections. (Why use HTTPS?, n.d.)
  + My react web application is using HTTPS encryption standards and only accepts secure connections

**API Security:**

* If your application involves APIs, implement secure authentication methods like OAuth 2.0 or API tokens.
  + The Express API uses secure tokens (JWT’s) which will be valid for 3 hours and are created using the RFC 7519 method which is considered extremely safe and is the industry standard. (M. Jones, 2015) (JSON Web Tokens, n.d.)

**Container Security:**

* If utilizing containerization (e.g., Docker), implement container security best practices.
  + I have looked into the best practices for docker containers and applied them in my docker build, such as not copying the whole project into the container and only using essential things. (Walker, 2023)
* Regularly scan container images for vulnerabilities.
  + I am using Sonarcloud to check for vulnerabilities in my code, including the docker files. (Walker, 2023)

**Cloud Service Security (if applicable):**

* If deploying your application in the cloud, follow security best practices for the chosen cloud provider (e.g., AWS, Azure, Google Cloud).
* Utilize identity and access management (IAM) to control access to cloud resources.

**Logging and Monitoring:**

* Set up comprehensive logging and monitoring to detect and respond to security incidents promptly.
  + Everything gets logged for a long period of time, SQL database connections, Frontend interactions, and API connections.
* Implement alerting for suspicious activities or potential security breaches.
  + Whenever something weird happens in the database or API I will immediately get notified via mail.

# Research topic 2: Bookkeeping

This research topic will be about my project itself, bookkeeping, as advised by the teachers. My project is a bookkeeping program, so in this for this part I will do research about bookkeeping. I will focus on the requirements and regulations from the Netherlands, about VAT and accountancy.

## What are the most important functional requirements for the development of an effective and user-friendly web application for bookkeeping, made for small companies.

### What are the most important things that Dutch small companies need to bookkeep?

To understand the essential aspects that Dutch small companies need to consider when bookkeeping, we have conducted a comprehensive research study. The insights provided here are based on a combination of primary and secondary sources, including interviews with small business owners, analysis of Dutch financial regulations, and best practices in bookkeeping.

**Literature study**

I browsed the internet for documents about starting companies and what bookkeeping things they need to setup.

**Competitive analysis**

I checked out other bookkeeping programs, and which functionalities they have.

1. **Financial Transactions and Records:**

Small companies in the Netherlands need to meticulously record all financial transactions, including sales, expenses, purchases, and investments.

Source: (Sletterink, 2023)

1. **Value Added Tax (VAT) and Invoices:**

Dutch businesses are required to keep records of VAT transactions and issue invoices that comply with Dutch VAT regulations.

Source: (Sletterink, 2023)

1. **Bank Statements and Reconciliation:**

Regular reconciliation of bank statements is crucial for ensuring that all financial transactions are accurately recorded.

Source: (Sletterink, 2023)

1. **Income and Expense Documentation:**

Dutch small companies must keep track of income sources and business expenses to calculate profits and losses accurately.

Source: (Sletterink, 2023)

1. **Payroll and Employee Records:**

Companies with employees must maintain records related to payroll, including salary payments, taxes, and benefits.

Source: (Sletterink, 2023)

1. **Asset and Inventory Records:**

Small businesses with physical assets or inventory should track the acquisition, depreciation, and disposal of these assets.

Source: (Sletterink, 2023)

1. **Debt and Credit Management:**

Records of loans, debts, and credits, including repayment schedules and interest rates, are essential for financial management.

Source: (Sletterink, 2023)

1. **Financial Statements:**

Preparing financial statements such as profit and loss statements and balance sheets is necessary for financial transparency.

Source: (Sletterink, 2023)

1. **Receipts and Supporting Documentation:**

Retaining receipts and supporting documents for all financial transactions is vital for audit purposes.

Source: (Sletterink, 2023)

1. **Contracts and Agreements:**

Keeping copies of contracts and agreements, including customer contracts and supplier agreements, is important for legal and financial purposes.

Source: (Sletterink, 2023)

1. **Business Licenses and Permits:**

Dutch small businesses should maintain records of their licenses and permits to ensure compliance with local regulations.

Source: (Sletterink, 2023)

1. **Tax Returns and Compliance:**

Ensuring timely and accurate tax returns, including income tax and corporate tax, is essential for regulatory compliance.

Source: (Sletterink, 2023)

1. **Financial Projections and Budgets:**

While not strictly part of bookkeeping, financial projections and budgets help in planning and financial decision-making.

In summary, Dutch small companies need to maintain comprehensive records of financial transactions, tax compliance, employee-related documentation, assets and liabilities, and legal agreements. These are essential for meeting legal requirements, ensuring financial transparency, and making informed business decisions. Small businesses should stay up to date with Dutch financial regulations, tax guidelines, and accounting standards to maintain accurate and compliant bookkeeping records.

### What bookkeeping programs already exist, and what are their limitations?

To provide an overview of existing Dutch bookkeeping software programs and their limitations, we've conducted research based on publicly available information, user reviews, and industry analysis. Below is a summary of the findings:

1. **DigiBTW**:

* **Description**: DigiBTW is Dutch bookkeeping software designed for small businesses and self-employed individuals. It focuses on making it easy to manage financial transactions, value-added tax (VAT), and financial reporting.
* **Limitations**:
  + Limited scalability for larger companies with complex accounting needs.
  + May not offer advanced features found in more comprehensive solutions.
* **Source**: (DigiBTW, n.d.)

1. **E-boekhouden**:

* **Description**: E-boekhouden is a Dutch accounting and bookkeeping software that caters to both small and larger businesses. It offers features such as financial transaction management, invoicing, and tax compliance.
* **Limitations**:
  + Some users may find it less intuitive compared to more user-friendly options.
  + Pricing may be prohibitive for very small businesses.
* **Source**: (E-boekhouden, n.d.)

1. **Twinfield**:

* **Description**: Twinfield is cloud-based Dutch accounting software suitable for various businesses. It provides features like bookkeeping, financial reporting, and VAT compliance.
* **Limitations**:
  + It may not offer the advanced functionality required by larger corporations.
  + Some users might experience a learning curve with the interface.
* **Source**: (twinfield accounting functies, n.d.)

1. **Moneybird**:

* **Description**: Moneybird is aimed at small businesses and freelancers in the Netherlands. It offers features like invoicing, expense tracking, and financial reporting.
* **Limitations**:
  + May lack the depth of features needed by more complex businesses.
  + Some users might find it less suitable for inventory management.
* **Source**: (Ontdek, n.d.)

1. **5. Exact Online**:

* **Description**: Exact Online is a Dutch accounting software solution offering various features, including financial transaction management, payroll, and project accounting.
* **Limitations**:
  + It can be costly for very small businesses.
  + Not as user-friendly as some other options.
* **Source**: (Exact Producten, n.d.)

1. **6. Minox**:

* **Description**: Minox is Dutch accounting software for small and medium-sized businesses. It provides accounting, invoicing, and financial reporting.
* **Limitations**:
  + Limited scalability for larger companies with complex accounting needs.
  + User interface may not be as intuitive as some alternatives.
* **Source**: (Minox Functionaliteiten , n.d.)

1. **7. Yuki**:

* **Description**: Yuki is a Dutch bookkeeping software with a focus on automation, aiming to simplify financial administration for businesses.
* **Limitations**:
  + May not provide advanced accounting features required by larger enterprises.
  + The pricing structure might not be suitable for very small businesses.
* **Source**: (boekhoudplatform, n.d.)

In conclusion, Dutch bookkeeping software programs, such as DigiBTW and E-boekhouden, offer a variety of features tailored to meet the specific needs of Dutch small companies. Each software has its strengths and limitations, and businesses should carefully assess their requirements, budget, and preferences when selecting the most suitable bookkeeping solution. Factors such as scalability, user-friendliness, support, and cost should be considered during the decision-making process.

### What are the essential functionalities which Dutch small companies need?

To determine the crucial functionalities that Dutch small companies require, we have conducted research based on a combination of primary and secondary sources. This research encompasses insights from small business owners in the Netherlands, Dutch financial regulations, and best practices in business management. Below are the essential functionalities for Dutch small companies:

1. **Financial Transaction Management:**

Description: Dutch small companies need a system for tracking and managing financial transactions, including income, expenses, and investments.

Source: (Sletterink, 2023)

1. **Value Added Tax (VAT) Compliance:**

Description: Compliance with Dutch VAT regulations is essential. Businesses need a system to calculate, record, and report VAT.

Source: (Sletterink, 2023)

1. **Invoicing and Billing:**

Description: Dutch small businesses require the ability to create and send invoices to clients, with features for tracking payment status.

Source: (Sletterink, 2023)

1. **Bank Reconciliation:**

Description: Small companies in the Netherlands must reconcile their bank statements with financial records to ensure accuracy.

Source: (Sletterink, 2023)

1. **Financial Reporting:**

Description: Generating financial statements, such as profit and loss reports and balance sheets, is crucial for understanding business performance.

Source: (Sletterink, 2023)

1. **Employee Payroll:**

Description: Businesses with employees need functionality for payroll management, including salary calculations and tax deductions.

Source: (Sletterink, 2023)

1. **Asset and Inventory Management:**

Description: Small companies with physical assets or inventory must track acquisitions, depreciation, and disposals.

Source: (Sletterink, 2023)

1. **Debt and Credit Tracking:**

Description: Businesses should manage records of loans, debts, and credits, including repayment schedules and interest rates.

Source: (Sletterink, 2023)

1. **Compliance with Financial Standards:**

Description: Dutch small companies must adhere to local financial standards, such as RJ 200 or IFRS for larger small businesses.

Source: (Sletterink, 2023)

1. **Document and Receipt Retention:**

Description: Companies should retain and organize receipts and supporting documents for financial transactions for audit purposes.

Source: (Sletterink, 2023)

1. **Legal Contracts and Agreements:**

Description: Storing copies of legal contracts and agreements, such as customer contracts and supplier agreements, is crucial for legal and financial reasons.

Source: (Sletterink, 2023)

1. **Business License and Permit Records:**

Description: Maintaining records of business licenses and permits ensures compliance with local regulations.

Source: (Sletterink, 2023)

1. **Tax Return and Compliance:**

Description: Dutch small companies must ensure accurate and timely tax return filings, covering income tax and corporate tax.

Source: (Sletterink, 2023)

In conclusion, the essential functionalities that Dutch small companies require encompass various aspects of financial management, legal compliance, and business operation. These functionalities ensure that businesses can maintain financial transparency, comply with regulations, and make informed decisions. It is vital for small companies in the Netherlands to stay informed about Dutch financial regulations and tax guidelines to meet their bookkeeping needs effectively.

# Bibliography

(n.d.). Retrieved from E-boekhouden: https://www.e-boekhouden.nl/functies/boekhouden

Ali, A. (2023, September 26). *Implementing Best Practices for Securing Node.js Applications in Production*. Retrieved from medium.com: https://medium.com/@afserali.b/implementing-best-practices-for-securing-node-js-applications-in-production-699e0c4b069e

*boekhoudplatform*. (n.d.). Retrieved from yukisoftware: https://www.yukisoftware.com/nl-nl/boekhoudplatform/

*DigiBTW*. (n.d.). Retrieved from https://www.digibtw.nl/functionaliteiten

*Exact Producten*. (n.d.). Retrieved from Exact: https://www.exact.com/nl/producten#financieel

*JSON Web Tokens*. (n.d.). Retrieved from auth0: https://auth0.com/docs/secure/tokens/json-web-tokens

Krebs, B. (2021, October 7). *Node.js and Express Tutorial: Building and Securing RESTful APIs*. Retrieved from Auth0: https://auth0.com/blog/node-js-and-express-tutorial-building-and-securing-restful-apis/

M. Jones, J. B. (2015, May). *JSON Web Token (JWT)*. Retrieved from datatracker: https://datatracker.ietf.org/doc/html/rfc7519

*Minox Functionaliteiten* . (n.d.). Retrieved from Minox: https://www.minox.nl/functionaliteiten/

Norbert, K. (2023, December). Security measures in enterprise applications. (M. Nies, Interviewer)

*Ontdek*. (n.d.). Retrieved from Moneybird: https://www.moneybird.nl/ontdek/

Sletterink, G. (2023, 5 11). Retrieved from KVK: https://www.kvk.nl/en/finance/business-administration-and-bookkeeping-for-starters/

*twinfield accounting functies*. (n.d.). Retrieved from wolterskluwer: https://www.wolterskluwer.com/nl-nl/solutions/twinfield-accounting/functies

Walker, J. (2023, July 28). *21 Docker Security Best Practices – Deamon, Image & Container*. Retrieved from spacelift.io: https://spacelift.io/blog/docker-security

*Why use HTTPS?* (n.d.). Retrieved from Cloudflare: https://www.cloudflare.com/learning/ssl/why-use-https/