TABLE OF CONTENTS

[List of symbols and abbreviations 3](#_Toc625790295)

[Introduction 4](#_Toc1430068706)

[1 Topic 1 5](#_Toc527764518)

[1.1 Analysis of business process. 5](#_Toc1030791916)

[1.1.1 Problem Area Description 5](#_Toc769151863)

[1.1.2 Main Problems in Modern Software Systems 5](#_Toc1431290148)

[1.1.3 Overview of Existing Approaches and Their Limitations 6](#_Toc1729187389)

[1.1.4 Specific Statement of the Thesis Task 7](#_Toc556252072)

[1.1.5 Glossary of Key Concepts 7](#_Toc932258713)

[1.1.6 Problem Statement 7](#_Toc431500879)

[1.2 Alternative version of topic 1. 8](#_Toc182997434)

[1.2.1 Analysis of Business Processes (BP) – "AS-IS" Model 8](#_Toc441290734)

[1.2.2 AS-IS Process 1: Organizing an Event 8](#_Toc1512333247)

[1.2.3 AS-IS Process 2: Finding Christian Friends 10](#_Toc168436437)

[1.2.4 AS-IS Process 3: Finding a Christian Community in a New City 11](#_Toc527749913)

[1.2.5 TO-BE Process: Risen Web Application 12](#_Toc785325681)

[1.2.6 Improvements and Problem Resolution 14](#_Toc534226105)

[1.2.7 Conclusion 15](#_Toc109018628)

[2 Topic 2 16](#_Toc472529905)

[3 Topic 3 20](#_Toc415586729)

[4 Topic 4 22](#_Toc1149334259)

[Conclusions 23](#_Toc245486248)

[References 24](#_Toc571284249)

[Appendix A Example of horizontal arrangement of drawings 30](#_Toc174987206)

[Appendix B Example of an appendix with structural sections, tables and figures 31](#_Toc1104858499)

[Appendix B Example of the content of an explanatory note for a course project 35](#_Toc65697084)

# List of symbols and abbreviations

IHI – institution of higher education;

SPE – scientific and pedagogical employee;

DBMS – database management system;

SAS - software audit system;

UML - Unified Modeling Language.

PA - Problem Area.

SS - Software Systems.

# Introduction

Christianity is the most widely practiced religion in the world. 2.38 billion people identify themselves as Christian. Unity is a central value among Christians. However, nowadays churches and Christians specifically are not as united as became possible with the development of Internet communication Technologies. That is why having one place where believers can navigate events, exchange information, and manage communication between churches has become more relevant than ever before. It can help people to see that they are not alone and that there are other people who have similar views as they do.

The object of research is the system of social network that allows communication between people of similar worldviews.

The subject of research is the principles of exchanging, storing, and securing messages, generating a post feed, and the composition of notifications accompanying the processes of sending notifications about new activities in the web application.

The aim of the work is to provide the possibility of finding believers in a given city, organizing events and making a place for uniting churches from different denominations.

The task of the work consists in the development of the web application "Risen" intended for automating the communication between churches and believers around the world.

The application advances the way that people find each other and brings sense of unity.

# 1 Topic 1

## 1.1 Analysis of business process

### 1.1.1 Problem Area Description

The development of the web application "Risen" is aimed at automating communication processes within a unique problem area (PA): a social networking system designed to connect individuals and churches sharing similar worldviews, particularly those rooted in faith-based communities. This section analyzes the business processes of the target organization—a global network of believers and churches—identifies inefficiencies in the current implementation, and defines the problem to be addressed in this thesis.

The problem area encompasses a social network system that facilitates communication among people with aligned spiritual beliefs, enabling them to exchange messages, organize events, and foster unity across denominations. The primary activities include message exchange, secure storage of communications, generation of personalized post feeds, and delivery of notifications about new activities. These processes are intended to help users locate like-minded believers in specific geographic areas, coordinate gatherings, and strengthen inter-church collaboration. However, the absence of a tailored platform means that churches and believers currently rely on fragmented, generic tools that fail to meet their specific needs.

### 1.1.2 Main Problems in Modern Software Systems

Modern software systems (SS) used in this domain, such as general-purpose social media platforms (e.g., Facebook, WhatsApp) or church management software (e.g., Church Community Builder, Planning Center), face several challenges:

1. Lack of Specificity. Existing platforms are not designed to cater to the unique needs of faith-based communities seeking inter-denominational unity and localized believer discovery. They do not provide functionality for easy identification of belonging to a specific church and ministry of participation.
2. Security Concerns. Generic systems may not prioritize the privacy and security of sensitive spiritual discussions or personal data.
3. Limited Customization. Current solutions offer little flexibility for generating tailored feeds or notifications based on worldview alignment or geographic proximity. Specific concern are close sourced feed algorithms.

These shortcomings hinder the ability of churches and believers to connect effectively, underscoring the need for a specialized solution.

### 1.1.3 Overview of Existing Approaches and Their Limitations

Several technologies and approaches exist for building social networking or community-focused systems:

1. General Social Media Platforms (e.g., Facebook Groups). These provide broad connectivity but lack features for secure, worldview-specific communication or event coordination tailored to faith communities. Privacy concerns and irrelevant content dilute their effectiveness.
2. Church Management Software (e.g., Tithe.ly, Breeze). These tools focus on administrative tasks (e.g., donations, membership tracking) rather than social networking or believer discovery, limiting their scope for fostering unity across denominations.
3. Custom-Built Solutions. Some organizations develop bespoke systems, but these often suffer from high development costs, scalability issues, and a lack of modern features like real-time notifications or dynamic feeds.

While these approaches address parts of the problem, they fail to integrate the full spectrum of needs—secure messaging, localized discovery, and cross-denominational unity—into a single, user-friendly platform.

### 1.1.4 Specific Statement of the Thesis Task

The task of this thesis is to design and develop "Risen," a web application that automates communication between churches and believers worldwide. The application will enable users to find believers in a given city, organize events, and create a unified space for churches of different denominations. By addressing the identified shortcomings, "Risen" will advance how people connect based on shared faith and promote a sense of global unity.

### 1.1.5 Glossary of Key Concepts

Main term in this domain presented below.

1. Social Network System. A platform enabling interaction among users with shared interests or beliefs;
2. Believer Discovery. The process of identifying and connecting with individuals of similar worldviews in a specific location;
3. Message Exchange. The secure sending, receiving, and storing of communications between users;
4. Post Feed. A dynamically generated list of updates or content relevant to a user’s interests and connections;
5. Notifications. Alerts informing users of new activities, such as messages, events, or updates;
6. Inter-Denominational Unity. Collaboration and communication across different branches of a faith community.

### 1.1.6 Problem Statement

The general problem addressed in this thesis is the inefficiency and fragmentation of communication among faith-based communities due to the lack of a dedicated platform. By analyzing the current business processes and their shortcomings, "Risen" will provide a cohesive, secure, and tailored solution to enhance believer connectivity and church collaboration within the subject area of social networking for spiritual unity.

## 1.2 Alternative version of topic 1.

### 1.2.1 Analysis of Business Processes (BP) – "AS-IS" Model

Key activities for the faith based comunities in current systems include:

* Message Exchange. Believers and churches communicate using tools like email, WhatsApp, or Facebook Messenger;
* Event Organization. Churches plan gatherings or events, often coordinated via group chats and social media (e.g., Instagram);
* Believer Discovery. Individuals seek like-minded believers, typically through word-of-mouth, church bulletins, or social media groups;
* Content Sharing. Users post updates or spiritual content on platforms like Facebook or Instagram;
* Notifications. Users receive updates informally (e.g., manual posts or messages) rather than through automated alerts. Some churches use social media to communicate information with their congregations.

Given stakeholders can be mapped:

1. Users: Individual believers, church leaders, and administrators;
2. Tools: General-purpose platforms (e.g., Facebook, WhatsApp, Instagram) and church management software (e.g., Planning Center).

### 1.2.2 AS-IS Process 1: Organizing an Event

The first "AS-IS" process involves a user inviting people to a Christian event using WhatsApp to contact church members, posting an Instagram Story, and visiting other churches in person. The process begins with preparing an invite, followed by creating or selecting a WhatsApp group, sending messages, designing and posting an Instagram Story, and physically announcing the event at other churches. Responses are collected manually across platforms, and the user assesses whether enough people have been invited before confirming the event. Below is shown a bpmn diagram of the given process.

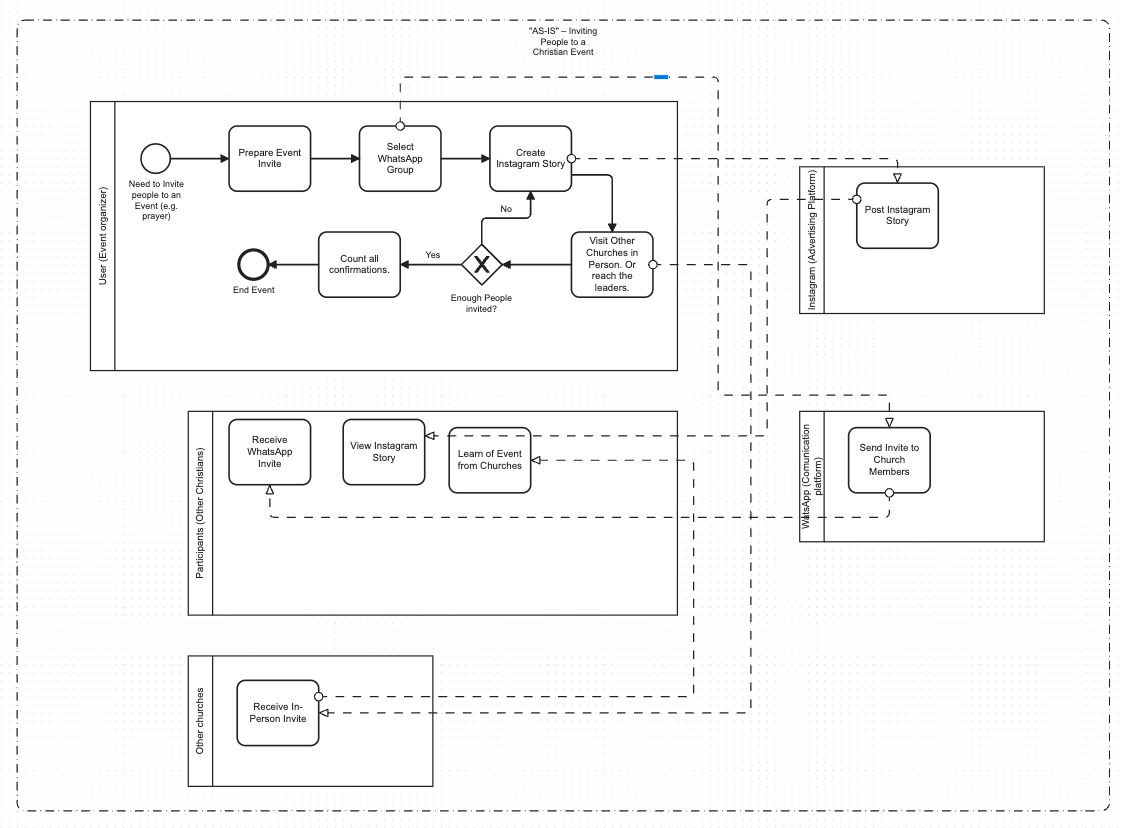


Figure 1.1 – "AS-IS". Inviting People to a Christian Event

Limitations:

1. Fragmentation. The user must manage three separate channels (WhatsApp, Instagram, in-person visits), leading to duplicate efforts and disjointed communication.
2. Manual Effort. Creating groups, posting Stories, and traveling to churches are time-consuming and labor-intensive.
3. Limited Reach. WhatsApp is restricted to known contacts, Instagram Stories expire after 24 hours, and in-person visits are geographically constrained.
4. Lack of Automation. Notifications and RSVP tracking are manual, increasing the risk of errors or oversights.

These inefficiencies hinder effective event coordination and limit the ability to foster unity across denominations.

### 1.2.3 AS-IS Process 2: Finding Christian Friends

The second "AS-IS" process describes a user seeking Christian friends via Instagram and Facebook. The user searches Instagram with hashtags (e.g., #ChristianLife), follows or messages potential friends, and explores Facebook groups by joining or posting introductions. Responses from both platforms are monitored, and the user evaluates whether enough connections have been made before building relationships. Below is shown a bpmn diagram of the given process.

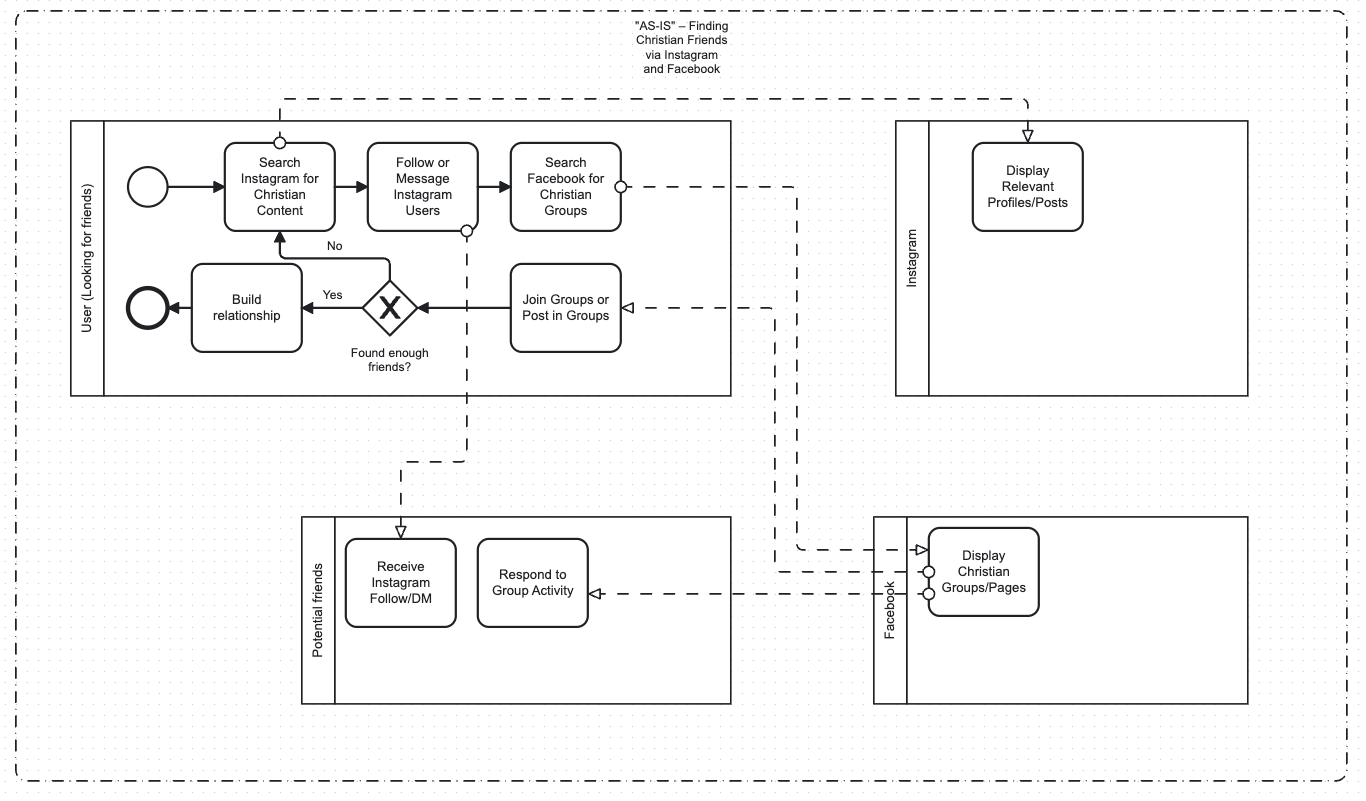


Figure 1.2 – "AS-IS" – Finding Christian Friends via Instagram and Facebook

Limitations:

1. Manual Search. Scouring hashtags and groups is repetitive and time-consuming, often yielding irrelevant results.
2. Fragmentation. Managing interactions across two platforms requires switching contexts and tracking responses separately.
3. Uncertain Outcomes. Connections depend on others’ responsiveness, and there’s no guarantee of shared values or proximity.
4. Privacy Risks. Public posts or group interactions may expose personal details to unintended audiences.

This process is inefficient and lacks the specificity needed for meaningful faith-based connections.

### 1.2.4 AS-IS Process 3: Finding a Christian Community in a New City

The third "AS-IS" process involves a user searching for a Christian community in a new city using Instagram and asking friends. The user searches Instagram for local churches (e.g., #[City]Church), follows or messages them, and contacts friends for recommendations. The user then researches suggested communities, assesses suitability, and connects with a chosen group. Below is shown a bpmn diagram of the given process.

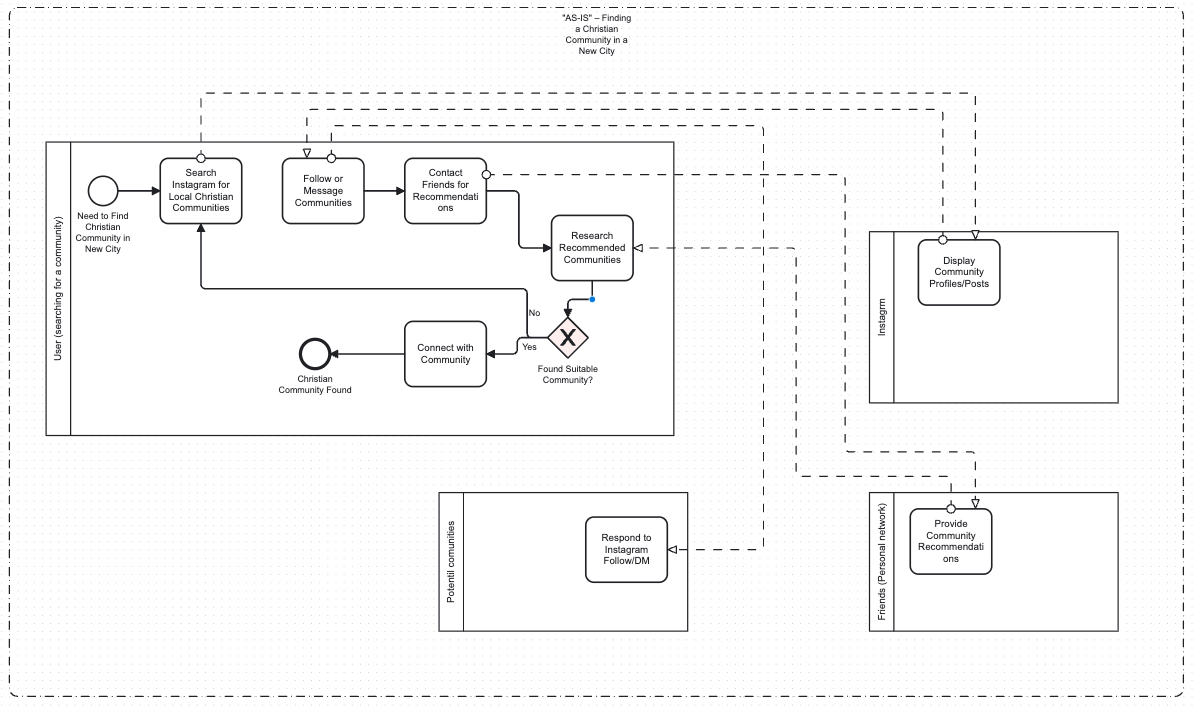


Figure 1.3 – "AS-IS" – Finding a Christian Community in a New City

Limitations:

1. Manual Effort. Searching Instagram and contacting friends individually are labor-intensive and prone to incomplete results.
2. Fragmentation. Combining Instagram results with friend recommendations requires manual consolidation, increasing complexity.
3. Limited Precision. Instagram searches may return outdated or irrelevant profiles, and friends may lack knowledge about the new city.
4. Inconsistent Responses. Community replies or friend suggestions vary in quality, delaying the connection process.

This fragmented approach makes it challenging for newcomers to integrate into a faith community efficiently.

### 1.2.5 TO-BE Process: Risen Web Application

The "TO-BE" BPMN diagram for the "Risen" web application unifies these processes into a single, streamlined workflow. Users log into "Risen," access a personalized dashboard, and select an action: find believers/friends, find a community, or organize an event. Each action is handled by a subprocess within the platform, leveraging secure messaging, automated notifications, and tailored search capabilities. The subprocesses—Find Believers/Friends, Find Community, and Organize Event—are modular, allowing users to pursue multiple goals concurrently. Responses are centralized, and users assess whether their goals are met before engaging further or selecting new actions. Below is shown a bpmn diagram of the given “TO-BE” process.

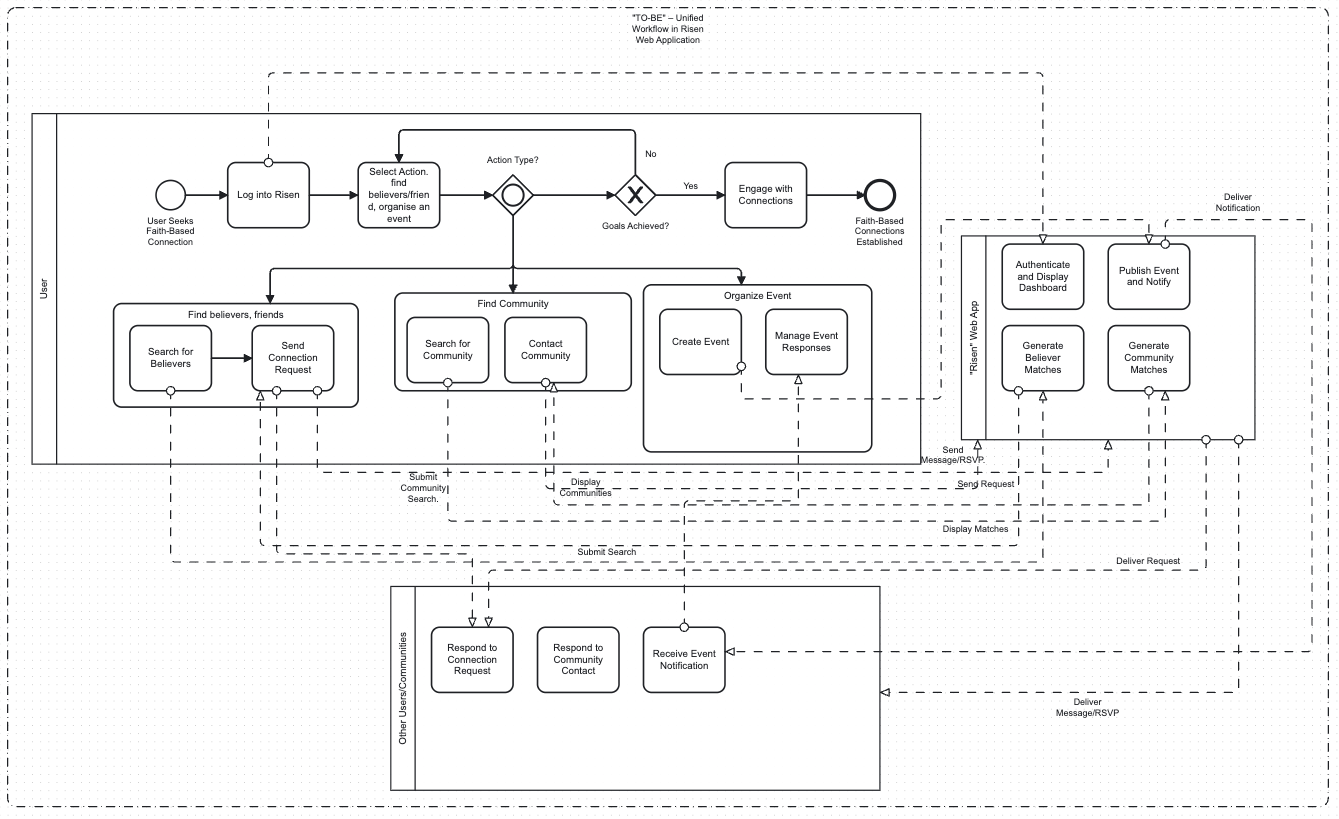


Figure 1.4 – "AS-IS" – Finding a Christian Community in a New City

Key Features of Risen:

* Believer Discovery Module. Users search for like-minded individuals using filters for location, denomination, and interests, powered by an open-source algorithm for transparency.
* Community Search Tool. Verified church profiles are returned based on user criteria, with direct messaging and event RSVP options;
* Event Management System. Users create events, publish them to targeted feeds, and receive automated notifications and RSVPs;
* Secure Messaging. End-to-end encryption ensures privacy for sensitive discussions;
* Personalized Feeds and Notifications. Dynamic feeds prioritize relevant content, and real-time alerts keep users informed.

### 1.2.6 Improvements and Problem Resolution

The "TO-BE" process in "Risen" significantly enhances the user experience and addresses the shortcomings of the "AS-IS" processes:

1. Centralization. Unlike the fragmented use of WhatsApp, Instagram, Facebook, and in-person visits, "Risen" consolidates all activities into one platform. Users no longer need to switch between tools or manually track responses, reducing complexity and saving time.
2. Automation. Manual tasks like sending invites, posting Stories, or visiting churches are replaced by automated features. For example, event creation in "Risen" triggers instant notifications to relevant users, and RSVPs are tracked centrally, eliminating the need for manual follow-ups.
3. Specificity and Relevance. The platform’s search filters ensure precise matches for believers and communities, unlike the broad, often irrelevant results from Instagram hashtags or Facebook groups. Users can find local Christians or churches aligned with their worldview, fostering meaningful connections.
4. Security. "Risen" prioritizes privacy with encrypted messaging and user-controlled settings, addressing concerns about public posts or insecure platforms like WhatsApp group chats. This creates a safe space for spiritual discussions.
5. Scalability and Reach. Unlike the limited reach of WhatsApp contacts or in-person visits, "Risen" enables global and local connections. Events can target specific denominations or geographic areas, and community searches span cities worldwide, promoting inter-denominational unity.
6. User Experience. The intuitive dashboard and modular subprocesses make "Risen" user-friendly. Users can seamlessly switch between finding friends, joining communities, and organizing events, with personalized feeds keeping them engaged. This contrasts with the disjointed, labor-intensive "AS-IS" processes.

### 1.2.7 Conclusion

The "AS-IS" processes for organizing events, finding Christian friends, and locating communities reveal significant inefficiencies: fragmentation, manual effort, limited specificity, privacy risks, and restricted reach. These shortcomings hinder the ability of Christians to connect and collaborate effectively, undermining the value of unity central to their faith. The "TO-BE" process in the "Risen" web application resolves these issues by offering a centralized, automated, and secure platform tailored to faith-based needs. By streamlining believer discovery, community engagement, and event coordination, "Risen" empowers users to build meaningful connections, foster inter-denominational collaboration, and experience a sense of global unity, fulfilling its mission to advance how Christians connect in the digital age.

# 2 Topic 2

In topic 2, it is proposed to present the results of detailed software design. The results include UML diagrams that allow us to present the behavior and structure of the software being created, as well as the results of designing a database for storing information (if this is provided for by the project):

This topic can have the following names:

2 Designing <software name>

2 Modeling <software name>

An alternative structure for this section's subdivisions is presented below.

2.1 Developing behavior diagrams … (or behavior modeling)

2.1.1 Developing use case diagrams for different user categories

2.1.2 Clarification of use case implementation

2.2 Developing a database for storing information

2.2.1 Defining domain business rules

2.2.2 Database structure development

When presenting the materials in this section, formulas, illustrations, and tables may appear in the text.

Formulas are placed along the text or on separate lines. Simple formulas are placed along the text, and basic formulas used in calculations and research are placed on separate lines. Only one formula can be placed on one line. Formulas are placed symmetrically to the text; one free line is left above and below each formula. The intervals between formulas that follow one another should be the same as in the text.

Explanations of the symbols of quantities and numerical coefficients, if they are not explained earlier in the text, should be given directly below the formula on a new line from the word "where" without a colon in the order in which they are given in the formula; a comma is placed after the formula. The interval between the formula and the explanation and between the explanation and the following text should be the same as in the text.

For example, information about objects to be represented in a database can be represented by tuples similar to the one shown below:

,

where – computer code;

– name of a software;

– processor model;

– processor power in GHz;

– amount of RAM, GB;

– hard disk space, GB;

– amount of available memory on the system disk, GB;

– the amount of available disk memory for users, GB.

This expression represents the elements of a computer description in the information system for accounting for computer classroom equipment.

The figures in the section may represent UML diagrams:

1. use case diagrams;
2. activity, sequence, and state diagrams that clarify use cases;
3. class and package diagrams.

In particular, the class diagram used to represent the structure of Entity classes used for object-relational mapping is shown in Fig. 2.1. The structure of database tables is also represented as a diagram (Fig. 2.2).

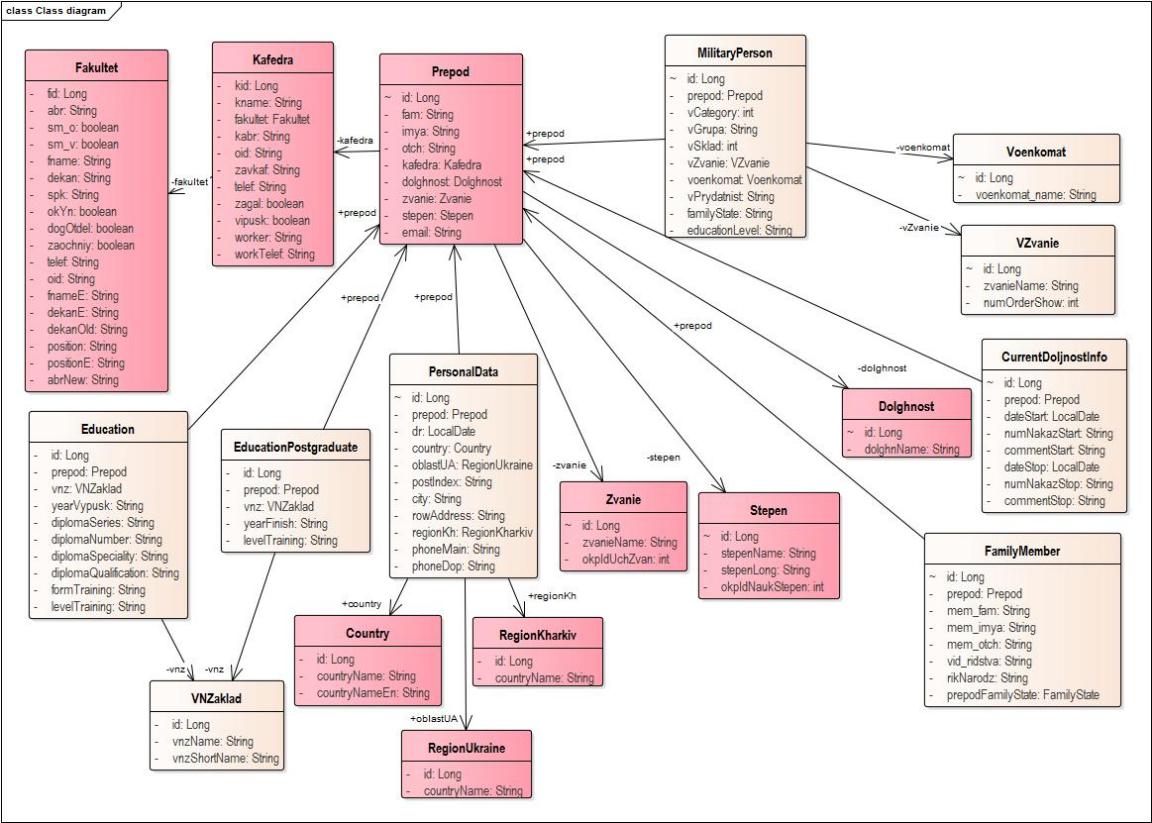


Figure 2.1 – Class diagram for creating a database

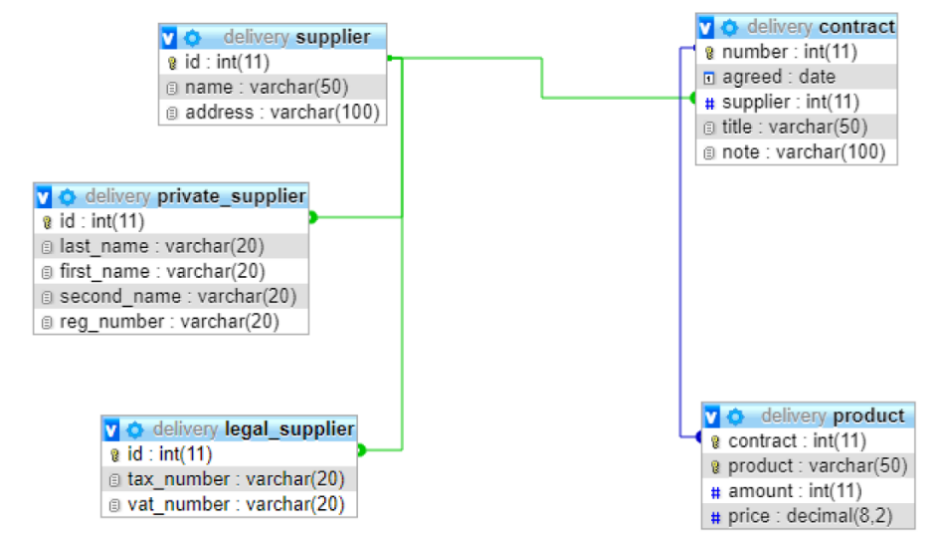


Figure 2.2 – Structure of delivery database tables

The tables in this section can be used to represent descriptions or attributes of entities, if an entity-relationship diagram (ERD) is provided (see Table 2.1), or attributes of classes, if the class diagram shown in Fig. 2.1 is provided (see Table 2.2), or columns of tables, if the structure of database tables is provided (see Table 2.3).

Table 2.1 – Description of the structure of the “Contracts” table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Key | Attribute name | Data type | Field size | Description |
| PK | contract\_ID | Numerical | Whole | contract number |
|  | supply\_date | Date/time | Short date format | delivery date according to the contract |
| FK | supplier\_ID | Numerical | Whole | supplier code |
|  | comment | Textual | Maximum | note |

Table 2.2 – Description of the fields “VNZaklad”

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Field name | Field description | Semantics |
| PK | id | @Id  @GeneratedValue(strategy = GenerationType.IDENTITY)  @Column(name = "vnz\_id")  private Long | Primary key |
|  | vnzName | @Column(name="vnz\_name")  private String | Full name of the educational institution |
|  | vnzShortName | @Column(length = 10, name="vnz\_short\_name",nullable = false,unique = true)  private String | Abbreviated name of educational institution |

Table 2.3 – Description of the structure of the “Contracts” table

|  |  |  |  |
| --- | --- | --- | --- |
| Key | Column name | Column data type | Semantics and description of input features |
| PK | number | int(11) | Contract number |
|  | agreed | date | Delivery date according to the contract. Required, format |
| FK | supplier | int(11) | Supplier code, required. Foreign key to the Supplier table |
|  | title | varchar(50) | "Hat" of the contract |
|  | note | varchar(100) | Note |

# 3 Topic 3

In topic 3, it is proposed to present the justification for the choice of a technology stack for software development (SW), as well as the principles of software testing both during creation and at the stage of assembly into a ready-made solution. To justify the choice of a technology stack and the format of a database management system (DBMS) for storing information, it is advisable to provide tables and graphs that reflect the data of the conducted studies. It should be noted that a justified choice involves the analysis of two or more possible options according to various criteria that arise from the requirements for the program. If such an analysis is not performed in the work (for example, the technology stack has already been defined by the team or it is determined by the technology that was used for components already existing in the system), then a DESCRIPTION of the technology stack should be presented, noting the features of the application, existing advantages and disadvantages.

The following list of divisions is proposed.

3.1 Choosing a software architecture

or

3.1 Choosing an application type

3.2 Choosing a database management system format

3.3 Choosing a programming language and frameworks

3.4 Definition of software development tools

3.5 Identifying Components and Developing a Deployment Diagram

This section may also include figures and tables. Fig. 3.1 presents statistics on the use of programming languages for backend development. Tab. 3.1 is a variant of presenting the results of the comparative analysis of cloud Java frameworks. It should be noted that both the figure and the table provide a clear justification for the choice of software development technologies, providing the ability to transfer the presentation to slides that will be used during the defense.

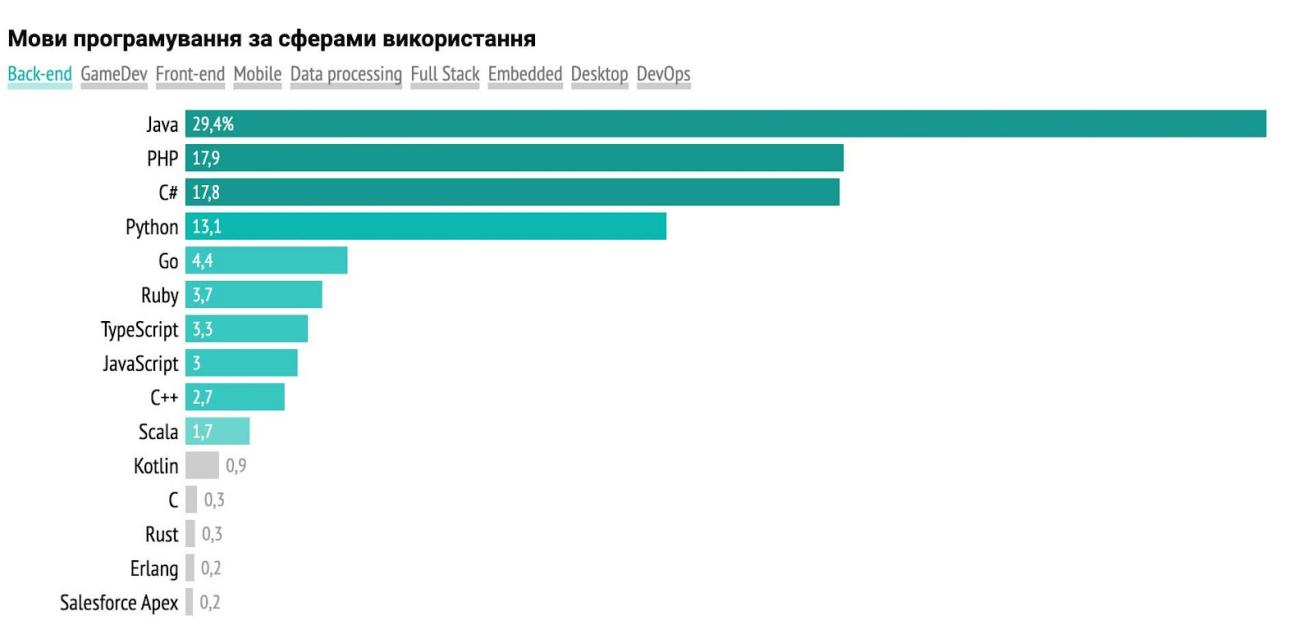


Figure 3.1 – Statistics on the use of programming languages ​​in backend development

Table 3.1 – Advantages and disadvantages of existing cognitologist’s workstation systems

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Web application name  Criterion | draw.io | LucidChart | Creately | Gliffy |
| Cost | Free | $85/month | $50/month | $100/month |
| Ease of use | 5 | 4 | 4 | 4 |
| Authentication | - | + | - | + |
| Project management for the network model | - | + | + | + |
| Creating a network model | + | + | + | + |
| Types of model storage | On the carrier | On media, database | On media, database | On media, database, cloud |
| Checking the correctness of the constructed network | - | - | - | - |
| Taking into account the rules for linking network vertices | - | - | - | - |

# 4 Topic 4

In topic 4 it is proposed to present:

1. the procedure for preparing the developed software (software) for use, in particular, the procedure for deploying the database, the composition of the installation package, the procedure for deploying the necessary components on the server;
2. instructions for different categories of users for using the developed software for its intended purpose.

The procedure for application is described in an arbitrary form. When presenting the description, it is advisable to focus on the appearance of usage diagrams for the relevant categories of users. It is also advisable to provide information about the features of data entry and commands on different pages (forms, screens) of the interface.

Based on the methodological guidelines, this section is optional if the student has only completed the development of a software project during the course design.

# Conclusions

The conclusions should contain the following elements:

1. reminders of the task being solved and the goal to achieve which work was done;
2. main results obtained during the work;
3. conclusions (at least three) that can be drawn from the perspective of software engineering issues that can be drawn from the results of software development;
4. practical value of the results obtained;
5. possible areas for improvement (continuation) of development or research (optional).

Requirements of STZVO-KhPI-3.01-2021

5.7 Conclusions

5.7.1 The conclusions should provide a summary of the results of the work performed and proposals for its use, as well as an assessment of the technical and economic efficiency of the work results and its implementation.

# References

1. STZVO-KhPI-3.01-2021. Text documents in the field of the educational process. General requirements for implementation. – Introduction. 09.12.2021.
2. DSTU GOST 7.1.:2006. Bibliographic record, bibliographic description. General requirements and rules of compilation. – Introduction. 01.07.2007.
3. Date K. J.. Introduction to database systems; translated from English / K. J. Date. - 8 editions - M. : Williams Publishing House, 2006. – 1328 p.
4. Knuth Donald Erwin. The art of programming: a classic work: in 3 volumes/D. E. Knout; community ed.; trans. with English Yu. IN. Kozachenko. - 2nd ed., ed. and additional - M. : "Williams" Publishing House, 2004.– Vol. 3. -823 p.
5. Tomashevsky V.M. System modeling: textbook. / V.M. Tomashevsky – K.: BHV Publishing Group, 2005. – 352 p.
6. Cybernetics in modern economic processes: collection of texts of speeches at the republican interuniversity scientific-practical conference / State Committee for Statistics of Ukraine, Institute of Statistics, Accounting and Auditing. – Kyiv: ISOA, 2002. – 147 p.
7. Pavlov A.A. Making decisions based on the hierarchical analysis method / A.A. Pavlov, E.I. Lyschuk // Vestn. NTU "Khpy". Collection of scientific works. Thematic issue "System analysis, management and information technologies" .– Kh.: NTU "KhPI" .– 2007 .– №41. - P.69-76
8. Titov P.S. Analysis and synthesis of dynamic processes in vibration devices for robots: master's thesis: 7.080303: protected 12.02.09 / Titov Petro Semenovich. – Kh., 2009. – 104 p. – I-13A.05.
9. Hnatush A. Reengineering: a lot in a little // http://citforum.ru/programming/case/gnatush/lot.shtml, 08.11.2018.
10. Kelton V. Simulation modeling. Classic CS. / V. Kelton. - 3rd ed. - St. Petersburg : Peter; K. : BHV Publishing Group, 2004. – 847 p.
11. The language of GPSS simulation modeling // http://gpssmodelling.narod.ru/index.html, 09/21/2018.
12. Gultyaev A.K. Matlab 5.2. Simulation modeling in the Windows environment: practice. manual / A. K. Gultyaev. - St. Petersburg : KORONA print, 1999. – 288 p.
13. Arena – simulation modeling system // http://www.interface.ru/sysmod/arena.htm, November 12, 2018.
14. A structured approach to enterprise modeling and analysis // http://www.idef.com/, 09/05/2018.

Requirements STZVO-KhPI-3.01-2021

Basic provisions

5.8.1 List of sources of information (SDI) is a list of cited, mentioned and used sources of information (SI). Sources of information are: books, articles, regulatory and technical documents (RTD), reports on scientific and research work, dissertations, technical and economic standards and norms, price lists, abstracts and reviews published as separate documents.

5.8.2 The list of sources of information includes the sources referenced in the text \*.

5.8.3 In the list of sources, bibliographic descriptions of sources of information are arranged in the order in which the sources are first mentioned in the text. The serial numbers of the descriptions in the list of sources of information are the numbers of references to them. An example of the execution of the source list is given in Appendix A. \* References are made in accordance with clause 6.3.10.4.

5.8.4 Bibliographic descriptions of information sources are given in the form in which they are presented in the information source (on the title page, the back of the title page and other elements of the document containing source and similar information) taking into account the requirements of DSTU GOST 7.1.

Example of a list of information sources

**References**

1 State certification system of Ukraine. Methods, rules, organization of activities: a reference book / Yu.I. Koifman, I.G. Kalman, O.Ya. Serdyukov. Kyiv: Lviv Publishing House, 1995.

2 Kurmaz L.V. Fundamentals of designing machine parts: a textbook. Kharkiv: Publishing house "Textbook of NTU "KhPI", 2010. 102 p.

. . . . . . . . . . .

9 Kolomoyets T. O. Administrative torts and administrative delicacy. Administrative law of Ukraine: textbook / edited by T. O. Kolomoyets. Kyiv, 2009. pp. 195–197.

10 Mykytiv G. V., Kondratenko Yu. Extratextual elements as a means of forming the media culture of readers of popular science magazines. Current problems of media education in Ukraine and the world: collection of abstracts of the additional international scientific and practical conference, Zaporizhzhia, March 3-4, 2016. Zaporizhzhia, 2016. pp. 50–53.

12 Vinichenko O. M. System of dynamic control of socio-economic development of an industrial enterprise: dissertation ... Doctor of Economic Sciences: 08.00.04. Dnipro, 2017. 424 p. . . . . . . . . .

16 DSTU 3582:2013. Bibliographic description. Abbreviations of words and phrases in the Ukrainian language. General requirements and rules. Effective from 22.08.2013.

APPENDIX B

Examples of bibliographic descriptions of sources of information

B.1 A bibliographic description of a source of information (SI) contains bibliographic information about it and consists of areas, the elements of which are given in a specific sequence using specific punctuation (grammatical punctuation marks and identification marks).

B.2 The bibliographic description of the CI can generally be presented in the form of a diagram:

Title proper: information relating to the title/statement of responsibility. – Publication information. – Specific information area. – Place of publication: Name of publisher, year of publication. – Issue number (for serials). – Physical characteristics area.

For a clear separation of areas and elements, spaces of one printed character are used before and after the proposed sign. The exceptions are the period and comma - spaces are left only after them. A period is placed at the end of the bibliographic description.

Elements of the bibliographic description and punctuation marks are submitted in accordance with DSTU GOST 7.1, taking into account the simplifications given in DSTU 8302, namely:

* the title provides information about one, two or three authors, while the names of these authors are not repeated in the bibliographic description in the information about responsibility (behind the slash);
* if necessary, more than three authors' names can be listed in the title;
* instead of the “period and dash” sign (“.–”), which separates the zones of the bibliographic description, it is recommended to use the “period” sign (within the same document, the use of punctuation marks is unified);
* information borrowed from other than the title page of the document may not be enclosed in square brackets;
* it is allowed not to indicate the general designation of the material after the title (“Text”, “Electronic resource”, “Maps”, “Music”, etc.);
* it is allowed not to provide the name (name) of the publisher in the source data;
* as part of the information about the physical characteristics of the document, you can indicate either its total volume (for example: 285 pp.), or the page number on which the reference object is presented (for example: P. 19);
* It is allowed not to provide information about the series and the International Standard Number (ISBN, ISMN, ISSN).

B.3 The title proper is given as it appears in the source of information.

Heading-related information contains information that reveals and explains the main heading, for example: textbook; reference book; plays for the theater, etc.

Liability information– this is information about individuals and organizations participating in the creation of the description object.

Publication informationcontain information about changes and features of this edition in relation to the previous one, for example, . – Fax. ed. ; . – Ed. 6th, corrected and supplemented. ; . – 10th ed.

In the area of ​​specific information, for example, the date of introduction and validity period for regulatory documents on standardization or the registration number of an application for a patent document, the date of its submission and publication, information about the official publication in which information about the patent document is published.

Place of publication and name of publisherare indicated as they are given in the source of information, for example, . – Kharkiv: Higher School; Kyiv: Publishing House "Sfera".

Year of publicationare given in Arabic numerals.

For serial publications, indicate the issue number by type:  
. – Issue 2; . – No. 3, etc.

Physical characteristic areacontains an indication of the physical form in which the object of the description is presented, in combination with an indication of the volume and, if necessary, its size, illustrations and accompanying material, for example,  
. - 8 tons ; . - 106 p. ; . - S. 11–19.

B. 6 In the case of a description of regulatory documents on standardization (standards and technical specifications), it is necessary to indicate the symbol, number of the standard (technical specifications), date of its entry into force and title, and for collections - the place of publication, name of the publisher and year of publication.

Examples

1. DSTU 3582:2013. Bibliographic description. Abbreviations of words and phrases in the Ukrainian language. General requirements and rules.  
   Effective from 08/22/2013.
2. System of occupational safety standards: collection. Kyiv: Publishing house of standards, 2002. P.102.

B.13 If the source of information is a document posted on the Internet, then the following scheme should be used when describing it:

Main heading: information related to the heading / information about responsibility // website address, date of page visit.

# Appendix A Example of horizontal arrangement of drawings

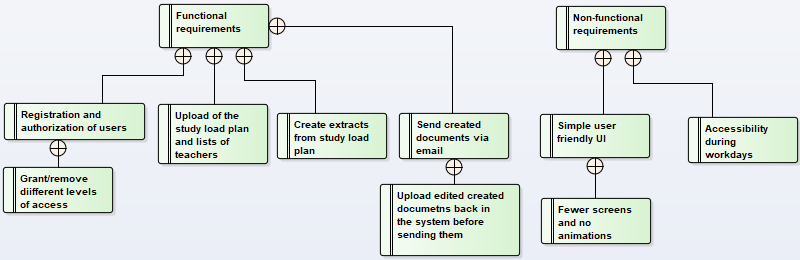


Figure B.1 – Diagram of requirements for a software solution for generating extracts from the distribution of the educational load

# Appendix B Example of an appendix with structural sections, tables and figures

B.1 Procedure for conducting the experiment

The sequence of actions when working with the program was described in more detail in the previous sections. The abbreviated version of the sequence is as follows:

1. upload a teaching workload plan for teachers;
2. download a list of teachers and detailed information about them;
3. download a list of teachers who are supervisors of students with course or diploma theses assigned to them;
4. click the button "Create a list of educational assignments";
5. click the "Create individual plans" button;
6. click the "Create workload distribution among teachers" button.

It was decided to additionally generate excerpts using the program not only on the same data that was used to generate excerpts without using the program, but also on specially prepared teaching load plans for different numbers of teachers to demonstrate the advantages for different sizes of departments.

B.2 Results of extract generation using the program

B.2.1 Measuring the time spent loading a plan

The symbols D will denote different conditional departments (English Department) with different numbers of teachers in each of them, numbers from 1 to 6 denote departments with 59, 33, 21, 17, 10 and 7 teachers respectively. The measurement of the time spent on loading the curriculum plan into the program in accordance with the number of teachers is given in Table B.1.

Table B.1 – Reading the curriculum plan by the program

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Experiment No. | D1 | D2 | D3 | D4 | D5 | D6 |
| 1 | 29.23 | 17.01 | 12.01 | 9.67 | 7.73 | 7.53 |
| 2 | 28.69 | 17.99 | 11.28 | 8.60 | 6.94 | 6.22 |
| 3 | 30.43 | 17.00 | 11.95 | 8.92 | 7.94 | 5.32 |
| 4 | 29.49 | 18.14 | 10.74 | 10.74 | 7.78 | 7.22 |
| 5 | 28.54 | 16.61 | 11.90 | 10.59 | 7.03 | 4.99 |
| 6 | 28.18 | 18.21 | 12.06 | 9.95 | 8.70 | 6.26 |
| 7 | 28.50 | 16.63 | 12.41 | 10.06 | 6.79 | 5.34 |
| 8 | 29.78 | 18.77 | 10.46 | 10.11 | 7.08 | 5.46 |
| 9 | 29.89 | 16.51 | 11.17 | 9.08 | 6.78 | 6.50 |
| 10 | 28.88 | 18.61 | 12.61 | 10.88 | 8.29 | 4.91 |
| Sum | 23.64 | 12.32 | 6.21 | 4.46 | 2.65 | 1.42 |

B.2.2 Measuring the time spent loading a plan

The reading measurements of the list of teachers are given in Table B.2.

Table B.2 – Time spent reading the list of teachers

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Experiment No. | D1 | D2 | D3 | D4 | D5 | D6 |
| 1 | 5.47 | 5.14 | 7.20 | 5.43 | 4.76 | 5.57 |
| 2 | 4.88 | 7.14 | 5.35 | 7.43 | 4.48 | 4.34 |
| 3 | 3.69 | 6.16 | 5.92 | 3.52 | 4.97 | 4.02 |
| 4 | 5.30 | 5.57 | 6.40 | 6.79 | 5.57 | 5.40 |
| 5 | 5.99 | 4.18 | 3.92 | 4.35 | 5.64 | 7.18 |
| 6 | 7.48 | 5.63 | 5.47 | 5.02 | 8.18 | 5.26 |
| 7 | 4.45 | 3.84 | 6.49 | 6.45 | 4.08 | 6.96 |
| 8 | 7.19 | 5.13 | 4.41 | 6.81 | 5.53 | 5.04 |
| 9 | 3.69 | 6.39 | 6.49 | 3.79 | 7.62 | 5.75 |
| 10 | 7.50 | 5.67 | 5.89 | 5.65 | 4.38 | 6.03 |

B.2.3 Measuring the time spent loading the list of teachers

The measurement of the time spent loading the list of teachers who are supervisors into the program is given in Table B.3.

Table B.3 – Time spent reading the list of lecturers and graduates

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Experiment No. | D1 | D2 | D3 | D4 | D5 | D6 |
| 1 | 6.37 | 3.96 | 4.11 | 4.28 | 5.71 | 6.22 |
| 2 | 4.28 | 6.25 | 4.05 | 4.44 | 5.61 | 3.61 |
| 3 | 5.01 | 6.41 | 4.16 | 4.06 | 6.26 | 5.10 |
| 4 | 5.75 | 4.76 | 6.01 | 6.18 | 5.19 | 5.46 |
| 5 | 4.26 | 3.93 | 5.04 | 5.26 | 4.46 | 4.86 |
| 6 | 5.53 | 3.90 | 6.37 | 4.53 | 5.68 | 5.29 |
| 7 | 4.62 | 3.53 | 4.43 | 4.76 | 4.37 | 4.19 |
| 8 | 5.90 | 3.57 | 3.60 | 5.49 | 4.70 | 4.17 |
| 9 | 5.68 | 4.81 | 5.87 | 5.72 | 4.19 | 4.09 |
| 10 | 4.04 | 4.28 | 3.59 | 4.42 | 4.49 | 5.80 |

…

B.5 Visualization of experimental results

Graphs of the dependence of the total time for forming excerpts on the number of teachers at the department are presented in Figures B.1 and B.2.

Figure B.1 – Graph of the dependence of the formation time on the number of teachers using the program

Evaluation of the obtained schedule allows us to state that the automated generation of extracts from the study load plan is carried out much faster, compared to the current approach to its generation. This conclusion is undoubtedly true for the department with 59 positions. The projected time for generating extracts from the study load distribution without using the program is approximately more than 10 hours. The developed application allows you to get a better result in less than 2 minutes.

# Appendix B Example of the content of an explanatory note for a course project

The appendix presents a version of the structure of the sections of the explanatory note for the course project using the example of work on the topic "Software design for the department's software audit system."

Introduction

List of symbols and abbreviations

Section 1. Justification of the need to automate the accounting of the department's software

1.1 Analysis of the current state of the department's software accounting organization

An analysis of the current state of business processes is presented with a detailed description and construction of a model of audience allocation planning processes, which uses information about computers and installed software, using BPMN.

A domain model of the subject area and a glossary have been developed.

1.2 Analysis of modern software audit systems with the development of proposals for implementation at the department

The current requirements of users interested in information about computer equipment in classrooms are described, changes to the accounting system are proposed, and a modified model of the classroom distribution planning process is presented, which takes into account equipment with computer technology using BPMN.

1.3 Development of a system of requirements for a software audit system

1.3.1 Description of the purpose of the system

1.3.2 Determining functional and non-functional requirements for the software being developed

1.3.3 Description of software development stages

Chapter 2. System Software Design

2.1 Development of behavior diagrams

2.1.1 Developing use case diagrams for different user categories

2.1.2 Refinement of use case implementation with development of interaction diagrams

2.2 Developing a database for storing information

2.2.1 Defining domain business rules

2.2.2 Database structure development

Chapter 3 Reasoned choice of technology stack for creating a system

3.1 Choosing a system architecture

3.2 Choosing a database management system format

3.3 Description of programming language and technologies for creating systems

3.4 Definition of software development tools

3.5 Identifying Components and Developing a Deployment Diagram

Chapter 4 Application of software for entering information about computers and department software

4.1 The work of the department's engineers during data entry

4.2 The work of teachers and information about the equipment of computer classes

4.3 Formation of the audience's IT passport

Conclusions

List of information sources