

**ДОДАТОК А**

Текст програми мобільного додатку

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
РАДІОЕЛЕКТРОНІКИ

«ЗАТВЕРДЖУЮ»

Керівник атестаційної роботи

проф. І. К. Сезонова

МОБІЛЬНИЙ ДОДАТОК ДЛЯ КЕРУВАННЯ ПІДПРИЄМСТВОМ

Текст програми мобільного додатку

АРКУШ ЗАТВЕРДЖЕННЯ

ГЮОИК.506190.020–01 12 01–А3

РОЗРОБИВ:

ст. гр. КТРСм–19–1

Меньшиков І.В.

## Б.1 – Main.java

```
public class Main {

    public static void main(String[] args) {
        LoginFrame app = new LoginFrame();
        app.setVisible(true);
    }
}
```

## Б.2 – LoginFrame.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class LoginFrame extends JFrame {
    private JButton loginButton = new JButton("Войти");
    private JTextField loginField = new JTextField(10);
    private JPasswordField passwordField = new JPasswordField(10);
    private JLabel loginLabel = new JLabel("Логин:");
    private JLabel passwordLabel = new JLabel("Пароль:");
    private DBconnection connection;

    public LoginFrame(){
        super ("Вход");
        this.setSize(500, 350);
        this.setLocationRelativeTo(null);
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new GridBagLayout());
        JPanel panel = new JPanel();
        panel.setLayout(new GridBagLayout());

        loginButton.addActionListener(e -> {
            connection = new DBconnection();
            try {
                Statement statement =
connection.getConnection().createStatement();
                ResultSet resultSet =
statement.executeQuery("select password from users where nickname
= '" + loginField.getText() + "'");
                resultSet.next();
                String correctPassword = resultSet.getString(1);
                String Password = passwordField.getText();
                if (Password.equals(correctPassword)) {
                    ModelingFrame modelingFrame = new
ModelingFrame();

```

```

        modelingFrame.setVisible(true);

LoginFrame.this.setDefaultCloseOperation(DISPOSE_ON_CLOSE);
        LoginFrame.this.processWindowEvent(new
WindowEvent(LoginFrame.this, WindowEvent.WINDOW_CLOSING));
    }
    else Errors.message("Неверный логин или пароль");
} catch (SQLException ex) {
    ex.printStackTrace();
    JOptionPane.showMessageDialog(null, "Неправильный
логин и/или пароль", "Ошибка", JOptionPane.ERROR_MESSAGE);
}
});

this.add(panel);
panel.add(loginLabel, new GridBagConstraints(0, 0, 1, 1,
1, 1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL, new
Insets(0, 0, 0, 15), 0,0));
panel.add(loginField, new GridBagConstraints(1, 0, 1, 1,
1, 1, GridBagConstraints.WEST, GridBagConstraints.HORIZONTAL, new
Insets(3, 0, 0, 0), 30,0));
panel.add(passwordLabel, new GridBagConstraints(0, 1, 1,
1, 1, 1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL,
new Insets(0, 0, 0, 15), 0,0));
panel.add(passwordField, new GridBagConstraints(1, 1, 1,
1, 1, 1, GridBagConstraints.WEST, GridBagConstraints.HORIZONTAL,
new Insets(3, 0, 0, 0), 0,0));
panel.add(loginButton, new GridBagConstraints(1, 2, 1, 1,
1, 1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL, new
Insets(3, 20, 0, 20), 0,0));

}
}

```

### Б.3 – ModelingFrame.java

```

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;

public class ModelingFrame extends JFrame{
    private Connection connection = new
DBconnection().getConnection();
    private JPanel panel = new JPanel();

    private void initComponents(){

        JButton wareHouseButton = new JButton("СКЛАД СЫРЬЯ");
        this.setLayout(new GridBagLayout());
    }
}

```

```

panel.setLayout(new GridBagLayout());
warehouseButton.setBackground(Color.WHITE);

//КОНВЕЙЕР А1

JLabel convA1Label = new JLabel("");
JLabel convA1label2 = new JLabel("Перемещается: ");
convA1Label.setPreferredSize(new Dimension(105, 20));

JPanel [] convA1 = new JPanel[15];
for (int i = 0; i < convA1.length; i++){
    convA1 [i] = new JPanel();
    convA1 [i].setBackground(Color.GRAY);
    panel.add(convA1[i], new GridBagConstraints(1, i + 1,
1, 1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
}
    new ConvThread(convA1, 6, convA1Label, 3, true, "Стекло");

    this.add(panel);
    panel.add(warehouseButton, new GridBagConstraints(0, 0,
10, 1, 1, 1, GridBagConstraints.NORTH,
GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0,0));
    panel.add(convA1Label, new GridBagConstraints(0, 7, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(convA1label2, new GridBagConstraints(0, 6, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));

//РЕЗЧИК СТЕКЛА

JButton cutglassButton = new JButton("ЧПУ станок для резки
стекла");
cutglassButton.setBackground(Color.WHITE);
JProgressBar cutglassBar = new JProgressBar();
panel.add(cutglassBar, new GridBagConstraints(0, 17, 2, 1,
1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
panel.add(cutglassButton, new GridBagConstraints(0, 16, 2,
1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
    new OrdersThread(cutglassBar, 7,6, 8, 4);

//КОНВЕЙЕР Б1

JLabel convB1Label = new JLabel("");
JLabel convB1label2 = new JLabel("Перемещается: ");
convB1Label.setPreferredSize(new Dimension(105, 20));

JPanel [] convB1 = new JPanel[15];
for (int i = 0; i < convB1.length; i++){
    convB1 [i] = new JPanel();

```

```

        convB1 [i].setBackground(Color.GRAY);
        panel.add(convB1[i], new GridBagConstraints(3, i + 1,
1, 1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
    }

    new ConvThread(convB1, 1, convB1Label, 2, true,
"Профиль");

    panel.add(convB1Label, new GridBagConstraints(2, 7, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(convB1label2, new GridBagConstraints(2, 6, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));

    //РЕЗЧИК ПРОФИЛЕЙ

    JButton cutprofButton = new JButton("ЧПУ станок для резки
профилей");
    cutprofButton.setBackground(Color.WHITE);
    JProgressBar cutprofBar = new JProgressBar();
    panel.add(cutprofBar, new GridBagConstraints(2, 17, 2, 1,
1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(cutprofButton, new GridBagConstraints(2, 16, 2,
1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 20, 0, 0), 0,0));
    new OrdersThread(cutprofBar, 2,1, 7, 3);

    //КОНВЕЙЕР Б2

    JLabel convB2Label = new JLabel("");
    JLabel convB2label2 = new JLabel("Перемещается: ");
    convB2Label.setPreferredSize(new Dimension(105, 20));

    JPanel [] convB2 = new JPanel[10];
    for (int i = 0; i < convB2.length; i++){
        convB2 [i] = new JPanel();
        convB2 [i].setBackground(Color.GRAY);
        panel.add(convB2[i], new GridBagConstraints(3, i + 18,
1, 1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
    }

    new ConvThread(convB2, 3, convB2Label, 9, false,
"Профиль");

    panel.add(convB2Label, new GridBagConstraints(2, 20, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(convB2label2, new GridBagConstraints(2, 21, 1,
1, 1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL,

```

```

new Insets(0, 0, 0, 0), 0,0));

    //ЧПУ ДЛЯ СВЕРЛЕНИЯ

    JButton sverlButton = new JButton("ЧПУ станок для
сверления");
    sverlButton.setBackground(Color.WHITE);
    JProgressBar sverlBar = new JProgressBar();
    panel.add(sverlBar, new GridBagConstraints(2, 30, 2, 1, 1,
1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(sverlButton, new GridBagConstraints(2, 29, 2, 1,
1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL, new
Insets(0, 20, 0, 0), 0,0));
    new OrdersThread(sverlBar, 5, 3,10, 10);

    //КОНВЕЙЕР БЗ

    JLabel convB3Label = new JLabel("");
    JLabel convB3label2 = new JLabel("Перемещается: ");
    convB3Label.setPreferredSize(new Dimension(105, 20));

    JPanel [] convB3 = new JPanel[10];
    for (int i = 0; i < convB3.length; i++){
        convB3 [i] = new JPanel();
        convB3 [i].setBackground(Color.GRAY);
        panel.add(convB3[i], new GridBagConstraints(3, i + 31,
1, 1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
    }

    new ConvThread(convB3, 8, convB3Label, 11, false,
"Профиль");

    panel.add(convB3Label, new GridBagConstraints(2, 36, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(convB3label2, new GridBagConstraints(2, 37, 1,
1, 1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));

    //ЧПУ ЗАЧИСТКИ ПРОФИЛЯ

    JButton weldingButton = new JButton("ЧПУ станок для
сварки");
    weldingButton.setBackground(Color.WHITE);
    JProgressBar weldingBar = new JProgressBar();
    panel.add(weldingBar, new GridBagConstraints(2, 42, 2, 1,
1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(weldingButton, new GridBagConstraints(2, 41, 2,
1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 20, 0, 0), 0,0));

```

```

new OrdersThread(weldingBar, 9, 8,12, 10);

//КОНВЕЙЕР B4

JLabel convB4Label = new JLabel("");
JLabel convB4label2 = new JLabel("Перемещается: ");
convB4Label.setPreferredSize(new Dimension(105, 20));

JPanel [] convB4 = new JPanel[5];
for (int i = 0; i < convB4.length; i++){
    convB4 [i] = new JPanel();
    convB4 [i].setBackground(Color.GRAY);
    panel.add(convB4[i], new GridBagConstraints(3, i + 43,
1, 1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
}

    new FinalConvThread(convB4, 10, convB4Label, "Профиль");

    panel.add(convB4Label, new GridBagConstraints(2, 44, 1, 1,
1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL, new
Insets(0, 0, 0, 0), 0,0));
    panel.add(convB4label2, new GridBagConstraints(2, 43, 1,
1, 1, 1, GridBagConstraints.SOUTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));

//КОНВЕЙЕР A2

JLabel convA2Label = new JLabel("");
JLabel convA2label2 = new JLabel("Перемещается: ");
convA2Label.setPreferredSize(new Dimension(105, 20));

JPanel [] convA2 = new JPanel[30];
for (int i = 0; i < convA2.length; i++){
    convA2 [i] = new JPanel();
    convA2 [i].setBackground(Color.GRAY);
    panel.add(convA2[i], new GridBagConstraints(1, i + 18,
1, 1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.VERTICAL,
new Insets(0, 0, 0, 0), 0,0));
}

    new FinalConvThread(convA2, 4, convA2Label, "Стекло");

// СБОРЩИК

JButton AssemblerButton = new JButton("Сборщик");
AssemblerButton.setBackground(Color.WHITE);
JProgressBar AssemblerBar = new JProgressBar();

    panel.add(AssemblerBar, new GridBagConstraints(0, 49, 10,
1, 1, 1, GridBagConstraints.NORTH, GridBagConstraints.HORIZONTAL,
new Insets(0, 0, 0, 0), 0,0));
    panel.add(AssemblerButton, new GridBagConstraints(0, 48,

```

```

10, 1, 1, 1, GridBagConstraints.NORTH,
GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0,0));

    //КОНВЕЙЕРНАЯ ЛИНИЯ В3

    JPanel [] convC1 = new JPanel[5];
    for (int i = 0; i < convC1.length; i++){
        convC1 [i] = new JPanel();
        convC1 [i].setBackground(Color.GRAY);
        panel.add(convC1[i], new GridBagConstraints(11 + i,
48, 1, 1, 1, 1, GridBagConstraints.NORTH,
GridBagConstraints.VERTICAL, new Insets(0, 0, 0, 0), 0,0));
    }

    new AssemblerThread(AssemblerBar, convC1);

    //СКЛАД ПРОДУКЦИИ

    JButton productWarehouseButton = new JButton("СКЛАД
ПРОДУКЦИИ");
    productWarehouseButton.setBackground(Color.WHITE);
    panel.add(productWarehouseButton, new
GridBagConstraints(16, 48, 10, 1, 1, 1, GridBagConstraints.NORTH,
GridBagConstraints.HORIZONTAL, new Insets(0, 0, 0, 0), 0,0));

}

private void initMenu(){
    JMenuBar menuBar = new JMenuBar();

    JMenu automatizationMenu = new JMenu("Автоматизация");
    JMenu orderMenu = new JMenu("Заказать");
    JMenu sellMenu = new JMenu("Склад");

    JMenuItem OrderFinalItem = new JMenuItem("Продукт");
    JMenuItem OrderDetailsItem = new JMenuItem("Сырьё");
    JMenuItem automatizationItem = new
JMenuItem("Управление");
    JMenuItem autocontrolItem = new JMenuItem("Настройка");
    JMenuItem sellItem = new JMenuItem("Продажа");
    JMenuItem prodItem = new JMenuItem("Продукция");
    JMenuItem detItem = new JMenuItem("Сырьё");

    orderMenu.add(OrderFinalItem);
    orderMenu.add(OrderDetailsItem);
    orderMenu.add(sellItem);
    automatizationMenu.add(automatizationItem);
    automatizationMenu.add(autocontrolItem);
    sellMenu.add(prodItem);
    sellMenu.add(detItem);

    menuBar.add(orderMenu);
    menuBar.add(automatizationMenu);

```

```

menuBar.add(sellMenu);

this.setJMenuBar(menuBar);

//слушаетали

OrderFinalItem.addActionListener(e -> {
    OrderFinal orderFinal = new OrderFinal();
    orderFinal.setVisible(true);
});

OrderDetailsItem.addActionListener(e -> {
    OrderDetailsFrame orderDetailsFrame = new
OrderDetailsFrame();
    orderDetailsFrame.setVisible(true);
});

automatizationItem.addActionListener(e -> {
    AutomaticFrame automaticFrame = new AutomaticFrame();
    automaticFrame.setVisible(true);
});

autocontrolItem.addActionListener(e -> {
    AutoControlFrame autoControlFrame = new
AutoControlFrame();
    autoControlFrame.setVisible(true);
});

sellItem.addActionListener(e -> {
    SellFrame sellFrame = new SellFrame();
    sellFrame.setVisible(true);
});

prodItem.addActionListener(e -> {
    ProductsFrame productsFrame = new ProductsFrame();
    productsFrame.setVisible(true);
});

detItem.addActionListener(e -> {
    DetailsFrame detailsFrame = new DetailsFrame();
    detailsFrame.setVisible(true);
});
}

public ModelingFrame(){
    super ("Моделирование рабочего процесса");
    this.setDefaultCloseOperation(EXIT_ON_CLOSE);
    this.setSize(1366, 768);
    this.setLocationRelativeTo(null);
    initComponents();
    initMenu();
}
}

```

## B.4 – DBConnection.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DBconnection {
    private final String URL =
"jdbc:mysql://localhost/enterprise_schema?" +
"user=root&password=root&serverTimezone=UTC";
    private Connection connection;
    public DBconnection(){
        try {
            connection = DriverManager.getConnection(URL);
        } catch (SQLException e) {
            System.out.println("SQLException: " + e.getMessage());
            System.out.println("SQLState: " + e.getSQLState());
            System.out.println("VendorError: " +
e.getErrorCode());
        }
    }
    public Connection getConnection() {
        return connection;
    }
}
```

## B.5 – ResultSetTableModel.java

```
import javax.swing.table.AbstractTableModel;
import java.sql.*;

public class ResultSetTableModel extends AbstractTableModel {
    private DBconnection DB;
    private Connection connection;
    private final Statement statement;
    private ResultSet resultSet;
    private ResultSetMetaData metaData;
    private int numberOfRows;

    public ResultSetTableModel (String query) throws SQLException{
        DB = new DBconnection();
        connection = DB.getConnection();
        statement =
connection.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,
ResultSet.CONCUR_READ_ONLY);
        resultSet = statement.executeQuery(query);
        metaData = resultSet.getMetaData();
        resultSet.last();
        numberOfRows = resultSet.getRow();
    }
}
```

```

        fireTableStructureChanged();
    }

    @Override
    public Class getColumnClass(int column) throws
IllegalStateException{
        try
        {
            String className = metaData.getColumnClassName(column
+ 1);
            return Class.forName(className);
        } catch (ClassNotFoundException ex) {
            System.out.println(ex.getMessage());
        } catch (SQLException e) {
            e.printStackTrace();
        }

        return Object.class;
    }

    @Override
    public int getColumnCount() throws IllegalStateException{
        try {
            return metaData.getColumnCount();
        } catch (SQLException sex) {
            System.out.println(sex.getMessage());
        }

        return 0;
    }

    @Override
    public String getColumnName(int column) throws
IllegalStateException
    {
        try {
            return metaData.getColumnName(column + 1);
        } catch (SQLException sex) {
            System.out.println(sex.getMessage());
        }

        return "";
    }

    @Override
    public int getRowCount() throws IllegalStateException
    {
        return numberOfRows;
    }

    @Override
    public Object getValueAt(int row, int column) throws
IllegalStateException

```

```

    {
        try {
            resultSet.absolute(row + 1);
            return resultSet.getObject(column + 1);
        } catch (SQLException sex) {
            System.out.println(sex.getMessage());
        }
        return "";
    }

    public void disconnectFromDatabase()
    {
        try {
            resultSet.close();
            statement.close();
            connection.close();
        } catch (SQLException sex) {
            System.out.println(sex.getMessage());
        }
    }
}

```

## B.6 – WarehouseFrame.java

```

import com.mysql.cj.x.protobuf.MysqlxCrud;

import javax.swing.*;
import javax.swing.border.Border;
import javax.swing.border.TitledBorder;
import javax.swing.event.ListSelectionEvent;
import javax.swing.table.DefaultTableCellRenderer;
import javax.swing.table.JTableHeader;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;

import static javax.swing.JOptionPane.showMessageDialog;

public class DetailsFrame extends JFrame {
    private JTable table = new JTable();
    private JTextField idTextfield = new JTextField(10);
    private JTextField nameTextfield = new JTextField(10);
    private JTextField amountTextfield = new JTextField(10);
    private Connection connection = new
DBConnection().getConnection();
    private int id = 0;

```

```

private void LoadRecords() throws SQLException {
    String sql_stmt = "SELECT * FROM details;";
    ResultSetTableModel tableModel = new
ResultSetTableModel(sql_stmt);
    table.setModel(tableModel);

table.getSelectionModel().addListSelectionListener((ListSelectionE
vent event) -> {
    try {
        if (table.getSelectedRow() >= 0) {
            Object detail_id =
table.getValueAt(table.getSelectedRow(), 0);
            Object name =
table.getValueAt(table.getSelectedRow(), 1);
            Object amount =
table.getValueAt(table.getSelectedRow(), 2);
            id = (int)
table.getValueAt(table.getSelectedRow(), 0);

            idTextfield.setText(detail_id.toString());
            nameTextfield.setText(name.toString());
            amountTextfield.setText(amount.toString());
        }
    } catch (Exception ex) {
        System.out.println(ex.getMessage());
    }
});

    DefaultTableCellRenderer rightRenderer = new
DefaultTableCellRenderer();
    rightRenderer.setHorizontalAlignment(SwingConstants.LEFT);

table.getColumnModel().getColumn(0).setCellRenderer(rightRenderer)
;
}

public void initComponents () {

    JLabel idLabel = new JLabel("Id:");
    JLabel nameLabel = new JLabel("Название:");
    JLabel typeLabel = new JLabel("Тип:");
    JLabel amountLabel = new JLabel("Количество");
    JPanel panel = new JPanel();
    JPanel panel2 = new JPanel();
    JPanel panel3 = new JPanel();
    JButton updButton = new JButton("Изменить");
    nameTextfield.setEditable(false);
    idTextfield.setEditable(false);

    updButton.addActionListener(updateAction);
    updButton.setBackground(Color.WHITE);
}

```

```
updButton.setCursor(Cursor.getPredefinedCursor(Cursor.HAND_CURSOR)
);
```

```

JScrollPane pane = new JScrollPane(table);
pane.setPreferredSize(new Dimension(450, 200));
panel2.add(pane, BorderLayout.SOUTH);

```

```

panel.setLayout(new GridBagLayout());
panel.setBorder(BorderFactory.createTitledBorder(null,
"Редактирование:", TitledBorder.DEFAULT_JUSTIFICATION,
TitledBorder.DEFAULT_POSITION, new Font("Tahoma", 1, 13)));
panel3.setLayout(new GridBagLayout());
this.setLayout(new GridBagLayout());
try {
    LoadRecords();
} catch (SQLException e) {
    e.printStackTrace();
}
this.add(panel3);
panel3.add(panel, new GridBagConstraints(0, 1, 1, 1, 1, 1,
GridBagConstraints.NORTH, GridBagConstraints.HORIZONTAL, new
Insets(0, 0, 0, 0), 0,0));
panel3.add(panel2, new GridBagConstraints(0, 0, 1, 1, 1,
1, GridBagConstraints.SOUTH, GridBagConstraints.HORIZONTAL, new
Insets(0, 0, 0, 0), 0,0));
panel2.add(pane);
panel.add(idLabel, new GridBagConstraints(0, 0, 1, 1, 1,
1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL, new
Insets(3, 0, 0, 0), 0,0));
panel.add(idTextfield, new GridBagConstraints(1, 0, 2, 1,
1, 1, GridBagConstraints.WEST, GridBagConstraints.HORIZONTAL, new
Insets(3, 0, 0, 0), 0,0));
panel.add(nameLabel, new GridBagConstraints(0, 1, 1, 1, 1,
1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL, new
Insets(3, 0, 0, 0), 0,0));
panel.add(nameTextfield, new GridBagConstraints(1, 1, 2,
1, 1, 1, GridBagConstraints.WEST, GridBagConstraints.HORIZONTAL,
new Insets(3, 0, 0, 0), 0,0));
panel.add(amountLabel, new GridBagConstraints(0, 3, 1, 1,
1, 1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL, new
Insets(3, 0, 0, 0), 0,0));
panel.add(amountTextfield, new GridBagConstraints(1, 3, 2,
1, 1, 1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL,
new Insets(3, 0, 0, 0), 0,0));
panel.add(updButton, new GridBagConstraints(0, 4, 1, 1, 1,
1, GridBagConstraints.EAST, GridBagConstraints.HORIZONTAL, new
Insets(3, 0, 0, 20), 0,0));
}

```

```
private ActionListener updateAction = new ActionListener() {
```

```

@Override
public void actionPerformed(ActionEvent e) {
    int dialogResult = JOptionPane.showConfirmDialog(null,
"Вы уверены, что хотите изменить эту запись?",
    "Подтвердите действие",
JOptionPane.YES_NO_OPTION);
    if (dialogResult == JOptionPane.YES_OPTION) {
        try {
            PreparedStatement preparedStatement =
connection.prepareStatement("update details set id_detail = ?,
name = ?, amount = ? where id_detail = ?");
            preparedStatement.setInt(1,
Integer.parseInt(idTextfield.getText()));
            preparedStatement.setString(2,
nameTextfield.getText());
            preparedStatement.setInt(3,
Integer.parseInt(amountTextfield.getText()));
            preparedStatement.setInt(4, id);
            preparedStatement.execute();
            LoadRecords();
        } catch (SQLException ex) {
            ex.printStackTrace();
        }
    }
};

public DetailsFrame() {
    super("CRUD операции");
    this.setDefaultCloseOperation(DISPOSE_ON_CLOSE);
    this.setSize(540, 480);
    this.setLocationRelativeTo(null);
    initComponents();
}
}

```

## Б.7 – ProductsFrame.java

```

import javax.swing.*;
import javax.swing.border.TitledBorder;
import javax.swing.event.ListSelectionEvent;
import javax.swing.table.DefaultTableCellRenderer;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;

```

```

import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class ProductsFrame extends JFrame {
    private JTable table = new JTable();
    private JTextField idField = new JTextField(10);
    private JTextField nameField = new JTextField(10);
    private JTextField amountField = new JTextField(10);
        private int id = 0;
        private Connection connection = new
        DBConnection().getConnection();
    private JTextField camersField = new JTextField();
    private JTextField profField = new JTextField();
    private JTextField glasswField = new JTextField();
    private JTextField heightField = new JTextField();
    private JTextField widthField = new JTextField();

private void LoadRecords (String stmt) throws SQLException{
    ResultSetTableModel tableModel = new
    ResultSetTableModel(stmt);
        table.setModel(tableModel);

table.getSelectionModel().addListSelectionListener((ListSelectionE
vent event) -> {
    try {
        if (table.getSelectedRow() >= 0) {
            Object productid =
table.getValueAt(table.getSelectedRow(), 0);
            Object name =
table.getValueAt(table.getSelectedRow(), 1);
            Object amount =
table.getValueAt(table.getSelectedRow(), 2);
            id = (int)
table.getValueAt(table.getSelectedRow(), 0);

```

```
idField.setText(productid.toString());
    nameField.setText(name.toString());
    amountField.setText(amount.toString());

    PreparedStatement preparedStatement =
connection.prepareStatement("SELECT camers, prof, glassw, height, width FROM
enterprise_schema.final_product where (id = ?);");
    preparedStatement.setInt(1, id);
    ResultSet resultSet = preparedStatement.executeQuery();
    resultSet.first();
    camersField.setText(Integer.toString(resultSet.getInt(1)));
    profField.setText(resultSet.getString(2));
    glasswField.setText(Integer.toString(resultSet.getInt(3)));
    heightField.setText(Integer.toString(resultSet.getInt(4)));
    widthField.setText(Integer.toString(resultSet.getInt(5)));
    }
} catch (Exception ex) {
    System.out.println(ex.getMessage());
}
});
DefaultTableCellRenderer rightRenderer = new DefaultTableCellRenderer();
rightRenderer.setHorizontalAlignment(SwingConstants.LEFT);
table.getColumnModel().getColumn(0).setCellRenderer(rightRenderer);
}
```

**ДОДАТОК Б**  
Демонстраційний матеріал

Харківський національний університет радіоелектроніки  
Кафедра  
комп'ютерно-інтегрованих технологій, автоматизації та мехатроніки"

Атестаційна робота магістра на тему:  
"Система керування підприємством за  
допомогою мобільного додатку"

Виконав:  
ст.гр. КТРСм-19-1  
Меньшиков Іван Володимирович

Керівник роботи:  
проф. Сезонова І.К.

## Мета роботи

*Об'єкт дослідження* – можливість використання мобільного додатку для підвищення ефективності діяльності підприємства

*Предметом* є впровадження мобільного додатку для збільшення обсягів виробництва та продажу

*Метою роботи* є розробка мобільного додатку для керування підприємством, дослідження його впливу на процес керування а також вивчення принципів розробки подібних додатків

## Основні завдання, які були виконані в роботі

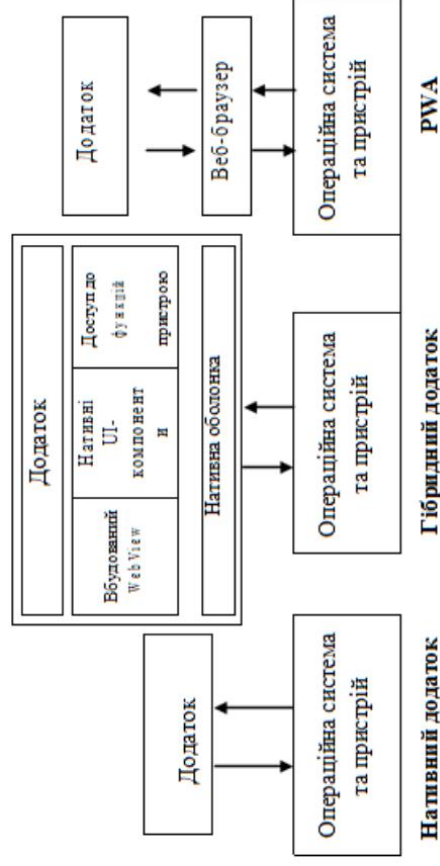
- уточнено зміст поняття «мобільний додаток» як складової інформаційної системи підприємства;
- аналіз основних типів мобільних систем та інструментів розробки мобільних додатків для різних платформ;
- визначення структури, функціональної значимості та принципів використання дистанційної системи керування підприємством;
- аналіз ринку МД для приладобудівних підприємств і їх основні характеристики;
- розробка мобільного додатку для приладобудівного підприємства.

# Поняття мобільного додатку

## Поняття мобільного пристрою

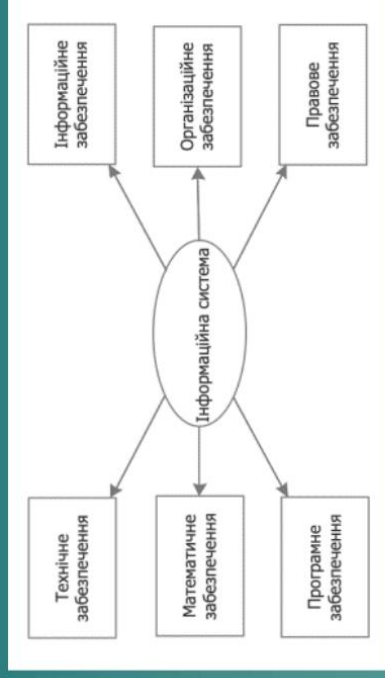
### Підходи до розробки додатків

	Нативні	Гібридні	PWA
Можливість перевикористання коду	Код розробляється окремо для кожної платформи	Можливе перевикористання коду	Можливе перевикористання коду
Доступ до функцій пристроїв	Найбільш повний	Обмежений доступ	Дуже низький
Модель розповсюдження	Завантаження в магазині додатків	Завантаження в магазині додатків	Доступ по URL
Продуктивність	Висока	Низька	Низька
Підтримка пристроїв	Висока	Висока	Середня
Популярність	Висока	Середня	Середня
Підтримка зовнішніх бібліотек	Висока	Середня	Висока

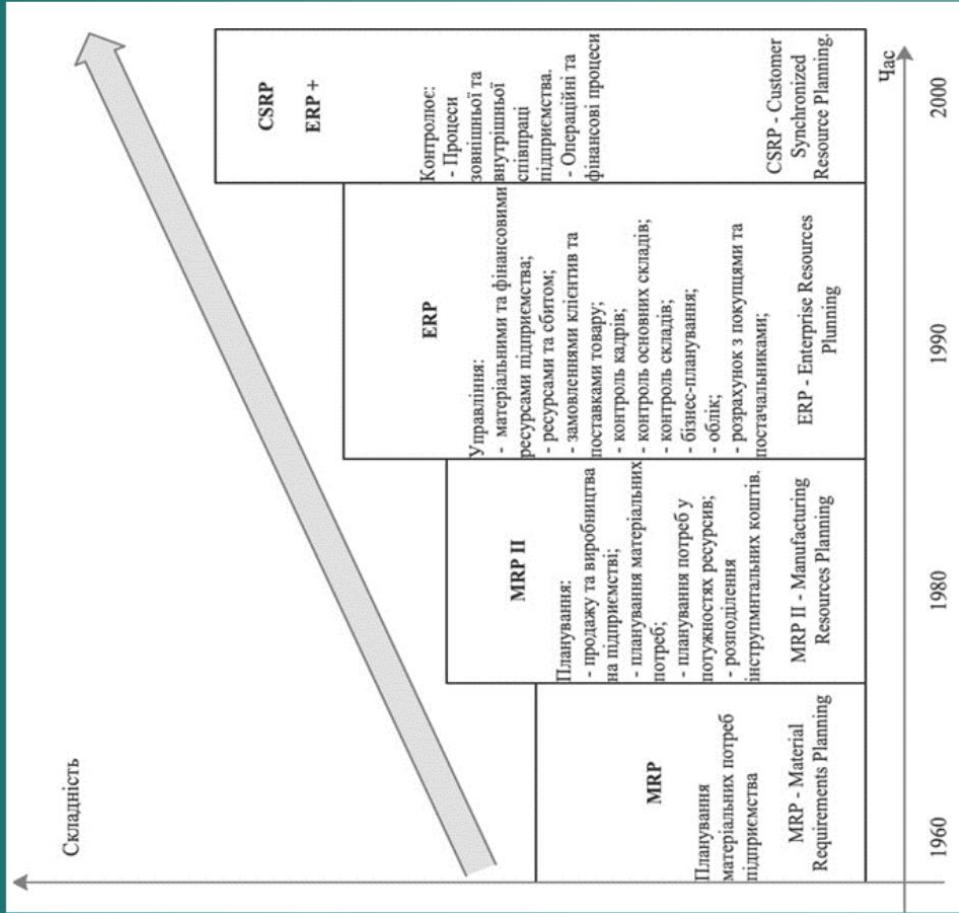


## Корпоративна інформаційна система

Корпоративна інформаційна система (КІС) – це інтегрована система управління підприємством або корпорацією, яка зумовлює використання усієї інформації (даних) про підприємство та їх поглиблений аналіз. Також вона працює з документообігом та електронних діловодством. Тобто, супроводжує повний цикл виробництва товару на підприємстві, з надходження матеріалів, виконання виробничого процесу, пересування та передачу замовнику.



# ERP - СИСТЕМА



Enterprise Resource Planning System – це система, яка дає можливість з часом додавати в неї будь – які модулі. Вона об'єднує в собі усі ресурси підприємства та має принцип єдиної бази даних.

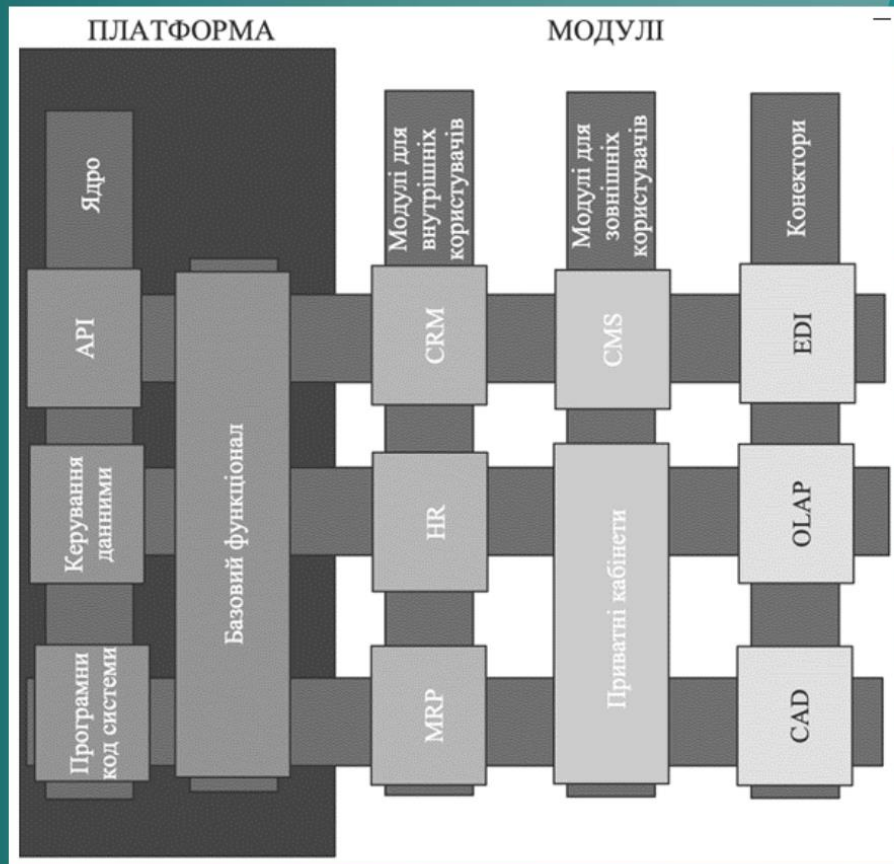
## ERP II:

- більш глибока функціональність;
- завдяки новим інструментам спростилися процес створення вузькоспеціалізованих галузевих рішень;
- міжкорпоративні бізнес–процеси удосконалилися та додалися нові моделі управління.

На основі викладеного визначимо основний функціонал дистанційної системи керування приладобудівним підприємством :

- формування плану продажів та планів виробництва;
- моніторинг потреб ресурсного забезпечення для підприємства;
- управління та контроль закупівлями та залишками товару на підприємстві;
- складання плану виробництва або поточного проекту;
- регулювання та оперативний контроль фінансової частини; контроль, моніторинг та управління проектами.

# Архітектура платформи та модулів ERP-системи

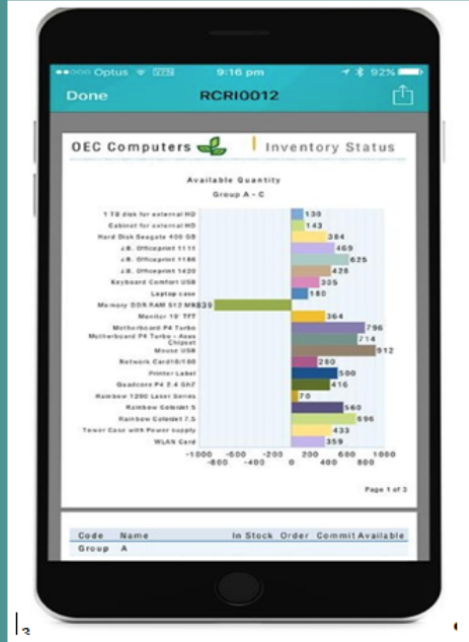


## Огляд існуючих рішень



ABAP (Advanced Business Application Programming, нім. Allgemeiner Berichtsaufbereitungs-Prozessor) – внутрішня мова програмування високого рівня німецької софтверної компанії SAP. Разом із Java є мовою створення додатків для SAP NetWeaver Application Server.

SAP є одним з провідних світових постачальників корпоративних програмних рішень як на території Германії, так і на території Європи. Вона організовує різні процеси всередині підприємства і між компаніями. Система включає в себе бізнес-додатки для великого і середнього бізнесу, а також стандартні рішення для малих і середніх компаній.



## CRM системи

Customer relationship management — поняття, що охоплює концепції, котрі використовуються компаніями для управління взаємовідносинами зі споживачами, включаючи збір, зберігання й аналіз інформації про споживачів, постачальників, партнерів та інформації про взаємовідносини з ними.

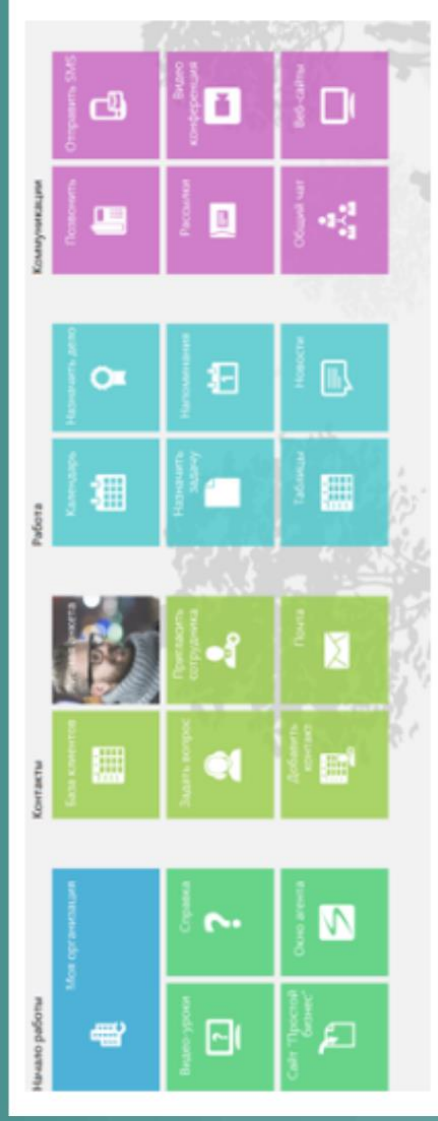
Концепція CRM-системи полягає в тому, що різноманітні інструменти ведення бізнесу об'єднуються в налагоджену систему. Замість таблиць Excel, месенджерів, багатьох документів та біганини по кабінетах залишається один-єдиний сервіс. У нього входять програми для збору даних про клієнтів, управління угодами, контролю за менеджерами, аналітики і прогнозування. Він спрощує рутину, прискорює прийняття правильних рішень і виключає помилки.

## CRM-система "Бізнес"

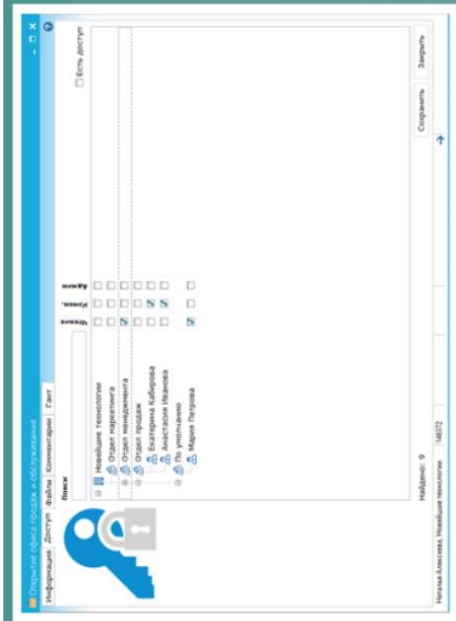
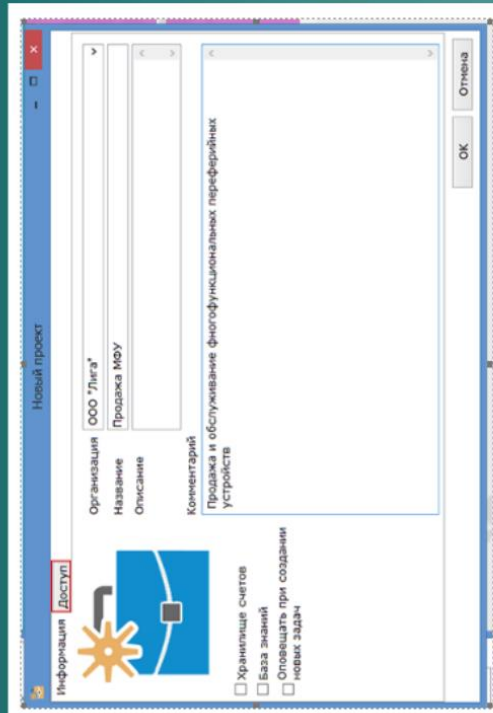
Спочатку треба виділити основні напрямки діяльності – проекти. Розбити їх на завдання. Призначити відповідальних за завданнями, встановити терміни, бюджет. Необхідно врахувати зміни статусу завдання в процесі виконання.

Управління проектами в CRM «Бізнес» здійснюється дистанційно.

Для цього вводиться база контактів постачальників по проектам. Контакти та історія листування завжди під рукою. Швидкий зв'язок здійснюється прямо з бази: можна зателефонувати за допомогою IP-телефонії, відправити SMS, e-mail або факс.



# Робота з проектами в CRM-системі "Бізнес"



## ВИСНОВКИ

- Результати проведеного дослідження дають можливість визначити, що мобільний додаток – автономне програмне забезпечення, призначене для роботи на смартфонах, планшетах та інших мобільних пристроях, що встановлюється через ринки: портали, магазини, маркетплейси з метою оптимізації і вирішення завдань користувача.
- На сьогодні існує декілька підходів технічної реалізації додатків для мобільних пристроїв, а саме – нативні, гібридні, прогресивні (PWA). Серед найбільш використовуваних їх типів для бізнесу та клієнтів слід виокремити:
  - для автоматизації процесів; для підвищення продуктивності, колаборації і спільної роботи; для «продовження» on-line сервісів та мобільні додатки як картка постійного клієнта.
- Обґрунтовано, що ключовою перевагою використання мобільних додатків для підприємців є можливість ретаргетингу який передбачає створення потужного бренду та розширення ринків збуту за рахунок аналізу поведінки покупця та нагадування про невиконані завдання за допомогою push-повідомлень, використання інформації про вродобання клієнтів на основі історії пошуку і покупок для рекламної видачі, врахування даних служб геолокації, інтеграція з соціальними мережами.

