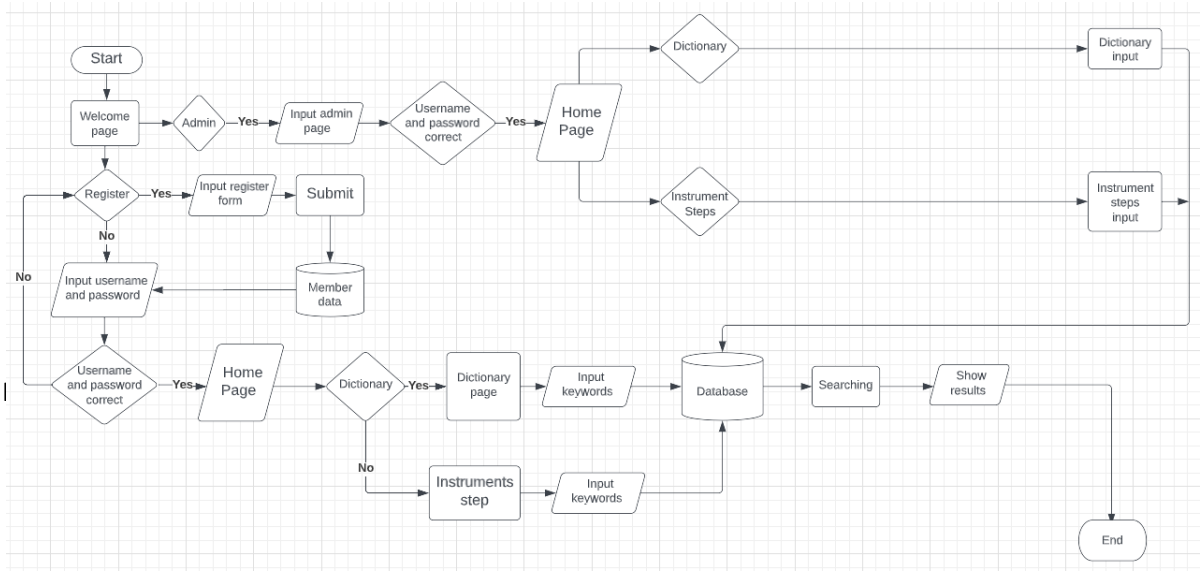
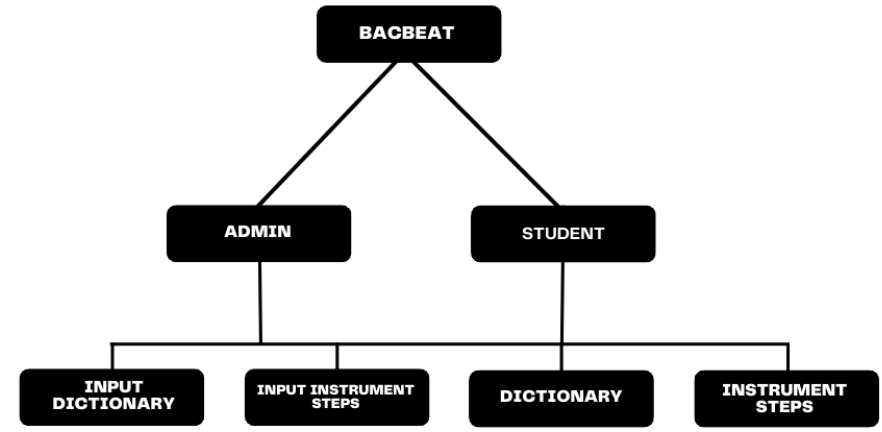
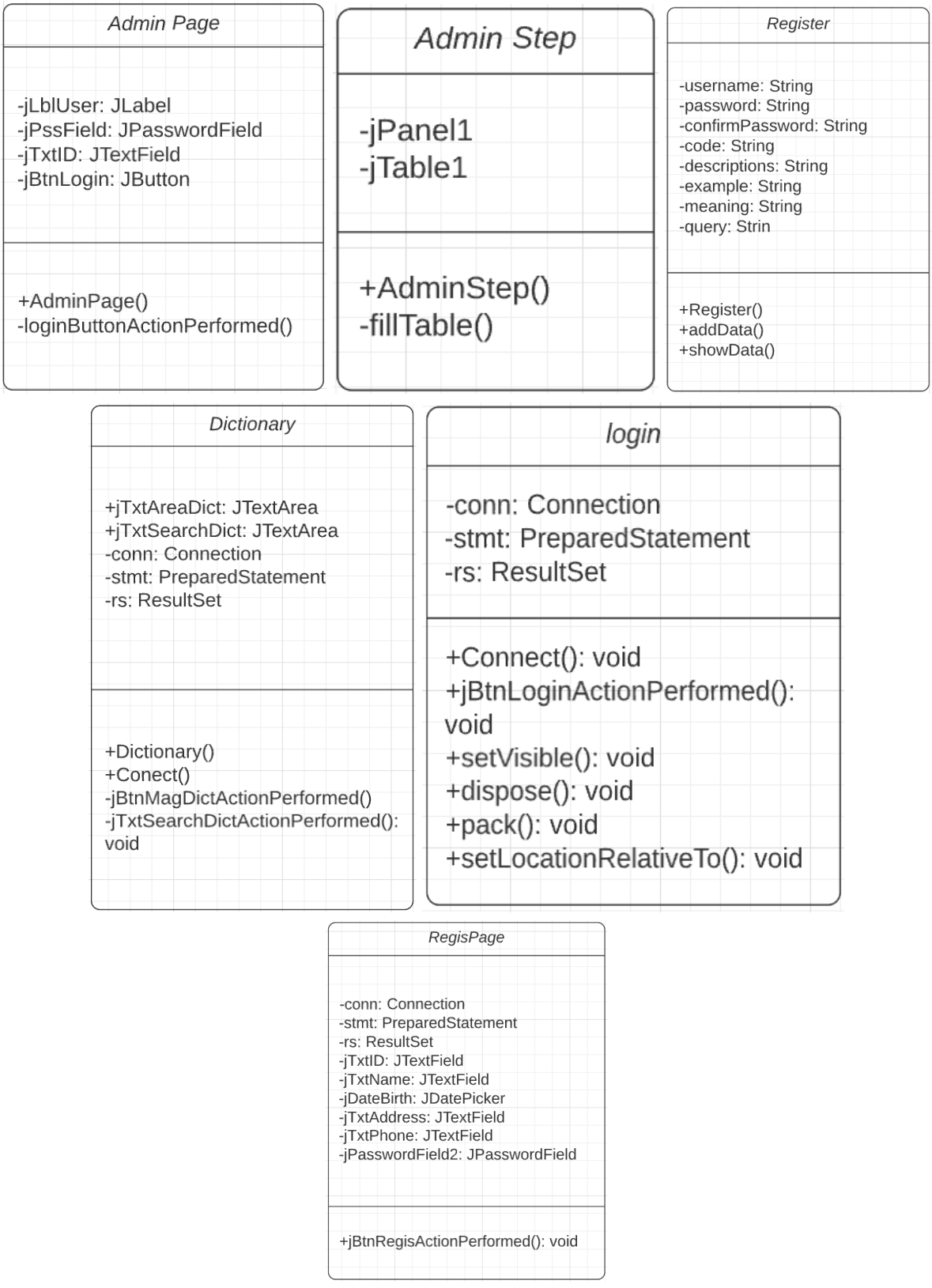
Criterion B: Design

Processing: Flowchart



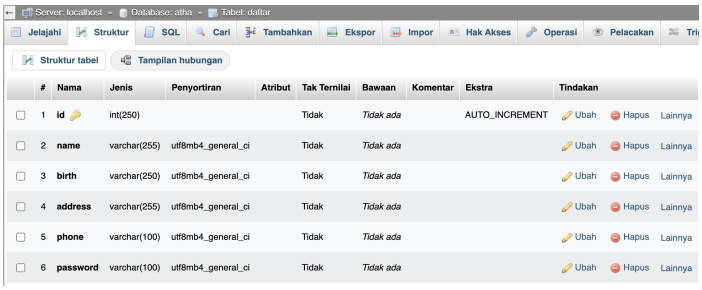
General View




Admin Page
-jLblUser: JLabel
-jPssField: JPasswordField
-jTxtID: JTextField
-jBtnLogin: JButton
+AdminPage()
-loginButtonActionPerformed()
Admin Step
-jPanel1
-jTable1
+AdminStep()
-fillTable()
Register
-username: String
-password: String
-confirmPassword: String
-code: String
-descriptions: String
-example: String
-meaning: String
-query: Strin
+Register()
+addData()
+showData()
Dictionary
+jTxtAreaDict: JTextArea
+jTxtSearchDict: JTextArea
-conn: Connection
-stmt: PreparedStatement
-rs: ResultSet
+Dictionary()
+Conect()
-jBtnMagDictActionPerformed()
-jTxtSearchDictActionPerformed(): void
login
-conn: Connection
-stmt: PreparedStatement
-rs: ResultSet
+Connect(): void
+jBtnLoginActionPerformed(): void
+setVisible(): void
+dispose(): void
+pack(): void
+setLocationRelativeTo(): void
RegisPage
-conn: Connection
-stmt: PreparedStatement
-rs: ResultSet
-jTxtID: JTextField
-jTxtName: JTextField
-jDateBirth: JDatePicker
-jTxtAddress: JTextField
-jTxtPhone: JTextField
-jPasswordField2: JPasswordField
+jBtnRegisActionPerformed(): void

Development Plan

Database

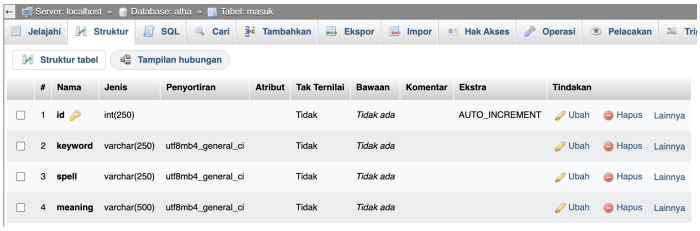

Server: localhost » Database: atha » Tabel: daftar

Jelajahi | Struktur | SQL | Cari | Tambahkan | Ekspor | Impor | Hak Akses | Operasi | Pelacakan

Struktur tabel | Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(250)			Tidak	*Tidak ada*		AUTO_INCREMENT	Ubah Hapus Lainnya
2	name	varchar(255)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
3	birth	varchar(250)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
4	address	varchar(255)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
5	phone	varchar(100)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
6	password	varchar(100)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya

(database for register and login)


Server: localhost » Database: atha » Tabel: masuk

Jelajahi | Struktur | SQL | Cari | Tambahkan | Ekspor | Impor | Hak Akses | Operasi | Pelacakan

Struktur tabel | Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(250)			Tidak	*Tidak ada*		AUTO_INCREMENT	Ubah Hapus Lainnya
2	keyword	varchar(250)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
3	spell	varchar(250)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
4	meaning	varchar(500)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya

(database for dictionary)

Server: localhost » Database: atha » Tabel: met

Jelajahi | Struktur | SQL | Cari | Tambahkan | Ekspor | Impor | Hak Akses | Operasi | Pelacakan

Struktur tabel | Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	id	int(250)			Tidak	*Tidak ada*		AUTO_INCREMENT	Ubah Hapus Lainnya
2	keyword	varchar(255)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
3	part	varchar(250)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya
4	step	varchar(500)	utf8mb4_general_ci		Tidak	*Tidak ada*			Ubah Hapus Lainnya

(database for instrument step)

The localhost has a database called “atha” that consists of three tables, “daftar” for the register and login data, “masuk” for the dictionary data, and “met” for the instrument steps data.

Schedule for Developing the Application

Program will be divided in two sections: Admin and Student sections; Admin section is used to input the dictionary and instrument steps into the database. Students can search the dictionary and instrument steps properly. I think 3 weeks for the admin section and 3 weeks for the user section.

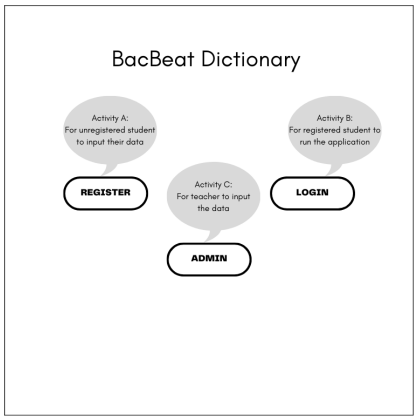
Admin Section	User Section
Create an interface	Create an interface
Write code for login	Write code for register, login, and restore in the database
Write code to add data (data will be located in database MySQL) for dictionary	Write code to create the search box to be able to search the database dictionary

Write code to add data (data will be located in database MySQL) for instrument steps	Write code to create the search box to be able to search the database instrument steps
Write code to display the data in the table from the database so it become for edit and delete	Write code to display data from database into the text area

Interface: Design the Application

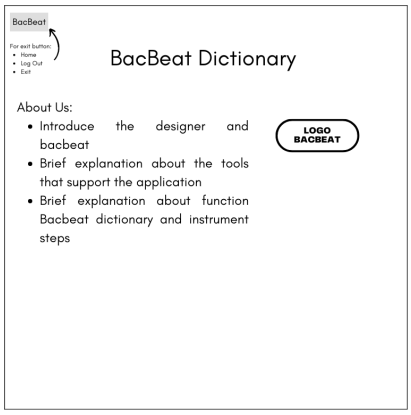
Activity A

Main Page



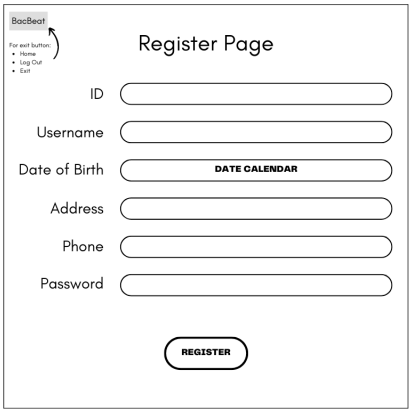
The user runs the application and enters the main page. Students who have not registered can press Register. Registered students can press Login. While the chosen student and the teacher can press Admin.

About Us



About us can be accessed by only registered students and admin because it is located inside the application.

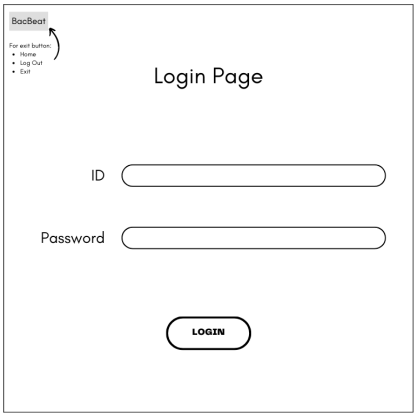
Register Page



On the Register Page, after filling in the data, the student can press the Register radio button so that the data can be entered into the database in table *daftar*.

Student Activity (Activity B)

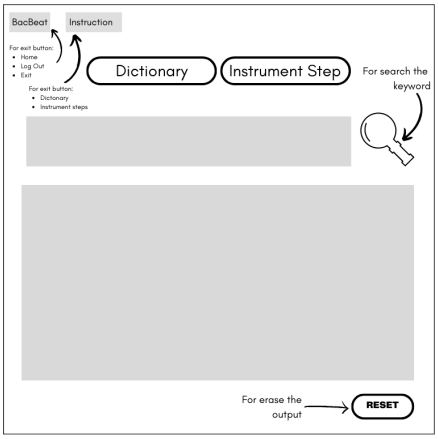
Login Page



On the login page, the application will adjust the input data from admin ID and password with the data stored in the database.

Input Data Table on the database - the student register data	Example
Location - jTextField	
Output data : The student with their data that has been recorded will display in the database table and it can be used for the login.	

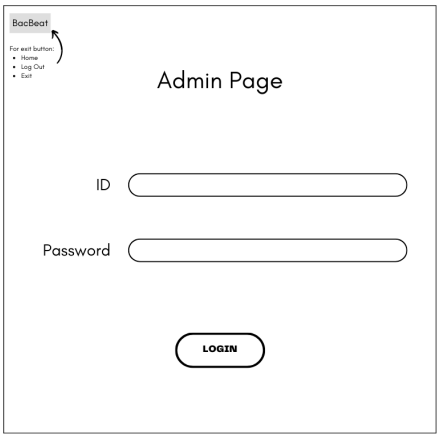
Home Page



The home page for students. There will be two tabs that each consist of jText area, jTextfield, and a magnifier button in each jForm to make it easier for students to explore the application.

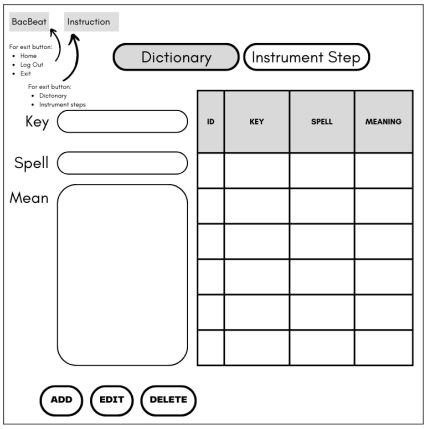
Admin Activity (Activity C)

Admin Page



For the admin page, the admin can input the name and password that has been determined, which is *admin* (name) and *bacbeat* (password) and presses the login radio button to enter the home page that is only for the admin.

Home Page



The home page for admin. The page form for the admin is specifically designed to input, edit, and delete data by selecting the radio buttons. The table on the right is used to make it easier for the admin to add, edit, or delete data from the database without having to go to the local host.

<table>
<tr>
<td>Input Data
Table - the dictionary and/or instrument steps data

Location - jTextField</td>
<td>Example
<table>
<tr><th>ID</th><th>Key</th><th>Spell</th><th>Meaning</th></tr>
<tr><td>1</td><td>Adagio</td><td>a-da-gi-o</td><td>a slow tempo</td></tr>
</table>
</td>
</tr>
<tr>
<td colspan="2">Output data : List data will display in the table the same as in the database.
Location - jTable</td>
</tr>
</table>

Test Plan

Action Test	Method of testing & Results
Test if the application runs and the main page appears with the radio button	Run the file, the main page is displayed and the radio buttons for admin, register, and login for users are working
Check if the register page works properly	Click on the “register” button on the main page and an appropriate page appears with the five text fields and one date chooser
Check if the admin login page works properly	Click on the “admin” button on the main page and an appropriate page appears with the name and password text field and login button
Check if the user login page works properly	Click on the “login” button on the main page and an appropriate page appears with the ID and password text field and login button
Check if the admin and user home page works properly	It works when the “login” button is pressed and it is displayed. Then, the radio button for the dictionary and instrument steps can show each of their pages when pressed.
Check if all the text fields can be filled by users and admin	If fields are not fields, especially password fill, alert messages should pop up and warn them
In the admin input page, both for the dictionary and instrument steps input page, the radio buttons for add, edit, and delete are working when mouse clicked	If add, the data will appear in the table. If edit, the data can be changed in the table. If delete, the data should not appear in the table

In the user page, both for dictionary and instrument steps, the radio button for the icon magnifier can be pressed.	Data from the database that is displayed via keywords from searches in text fields can be carried out by pressing the icon magnifier
The shortcuts for this application can execute their command when pressed	The program can run the application well.

Word Count : 314