

BAB 4

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap penterjemah perancangan berdasarkan hasil analisis ke dalam suatu bahasa pemrograman tertentu serta penerapan perangkat lunak yang dibangun pada lingkungan yang sesungguhnya. Implementasi dan pengujian sistem dilakukan bertujuan untuk mengetahui hasil dari sistem yang dibangun. Implementasi sistem dimulai dari penginstalan aplikasi oleh pengguna dan digunakan oleh pengguna untuk melakukan tanya jawab.

4.1.1 Implementasi Perangkat Keras

Implementasi perangkat keras merupakan spesifikasi perangkat keras yang digunakan dalam mengimplementasikan sistem yang dibangun. Dalam hal ini Perangkat keras yang diimplentasikan pada Pembangunan Aplikasi Pembelajaran Untuk Chatbot berbasis Android. Adapun perangkat keras yang diimplementasikan adalah sebagai berikut:

1. Perangkat Keras Komputer

Tabel 4.1 Perangkat Keras yang digunakan Pegawai

No	Perangkat Keras	Perangkat Keras Komputer untuk Membangun Sistem	Perangkat Keras Komputer Minimum untuk Menjalankan Sistem
1	Processor	Intel Core i3-3100M 2.40 GHz	Intel Dual Core 2.3 GHz
2	VGA	Nvidia Geforce GT 705M	Intel HD
3	RAM	DDR3 8GB	1GB
4	Hardisk	500 GB	1 GB

2. Perangkat Keras Smartphone Android

Tabel 4.2 Perangkat Keras yang digunakan Siswa

No	Perangkat Keras	Perangkat Keras Android untuk Membangun Sistem	Perangkat Keras Android Minimum untuk Menjalankan Sistem
1	Processor	Intel Atom 1.60 GHz x86	830 MHz ARMv6
2	OS	5.0.1 Lollipop	4.2 Jelly Bean
3	RAM	2 GB	512 MB
4	Storage	16 GB	512 MB

4.1.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak merupakan spesifikasi perangkat lunak yang digunakan dalam mengimplementasikan sistem yang dibangun. Dalam hal ini perangkat lunak yang dimaksud adalah sistem operasi yang digunakan oleh Pegawai dan Siswa. Implementasi perangkat lunak yang digunakan oleh Pegawai dan Siswa adalah sebagai berikut:

1. Perangkat Lunak Komputer

Tabel 4.3 Perangkat Lunak yang digunakan Pegawai

No	Nama Perangkat	Spesifikasi
1	Sistem Operasi	Windows 7 64 Bit
2	Web Browser	Mozilla Firefox, Google Chrome

2. Perangkat Lunak Smartphone Android

Tabel 4.4 Perangkat Lunak yang digunakan Siswa

No	Nama Perangkat	Spesifikasi
1	Sistem Operasi	Android Versi 5.0 Lolipop

4.1.3 Implementasi Antarmuka

Implementasi antarmuka bertujuan untuk mengubah hasil analisis menjadi bentuk nyata. Antarmuka merupakan bagian penting pada Pembangunan Aplikasi Pembelajaran Untuk Chatbot ini. Implementasi antarmuka pada pembangunan aplikasi ini terdiri dari antarmuka aplikasi pembelajaran untuk chatbot pada siswa

dan website pengolahan data untuk pegawai. Berikut rincian dari implementasi antarmuka yang di buat :

1. Implementasi Antarmuka Aplikasi *Backend*

Implementasi antarmuka Aplikasi *Backend* dapat dilihat pada tabel 4.5

Tabel 4.5 Implementasi Antarmuka Aplikasi Backend

No	Nama Antarmuka	Keterangan
1	Halaman <i>Login</i>	Menampilkan form untuk memasukan username dan password.
2	Halaman <i>Menambah Pengguna</i>	Menampilkan form untuk menambah data pengguna yang baru.
3	Halaman <i>Mengubah Pengguna</i>	Menampilkan form dari data pengguna yang akan diubah.
4	Halaman <i>Menghapus Pengguna</i>	Menampilkan data pengguna yang akan dihapus.
5	Halaman <i>Menambah Kategori Kursus</i>	Menampilkan form untuk menambah data kategori kursus yang baru.
6	Halaman <i>Mengubah Kategori Kursus</i>	Menampilkan form dari data kategori kursus yang akan diubah.
7	Halaman <i>Menghapus Kategori Kursus</i>	Menampilkan data kategori kursus yang akan dihapus.
8	Halaman <i>Menambah Kursus</i>	Menampilkan form untuk menambah data kursus yang baru.
9	Halaman <i>Mengubah Kursus</i>	Menampilkan form dari data kursus yang akan diubah.
10	Halaman <i>Menghapus Kursus</i>	Menampilkan data kursus yang akan dihapus.
11	Halaman <i>Upload Materi</i>	Menampilkan form untuk mengupload data materi yang baru.
12	Halaman <i>Logout</i>	Menampilkan tombol logout untuk keluar dari website.

2. Implementasi Antarmuka Aplikasi *Frontend*

Implementasi antarmuka Aplikasi *Frontend* dapat dilihat pada tabel 4.6

Tabel 4.6 Implementasi Antarmuka Aplikasi *Frontend*

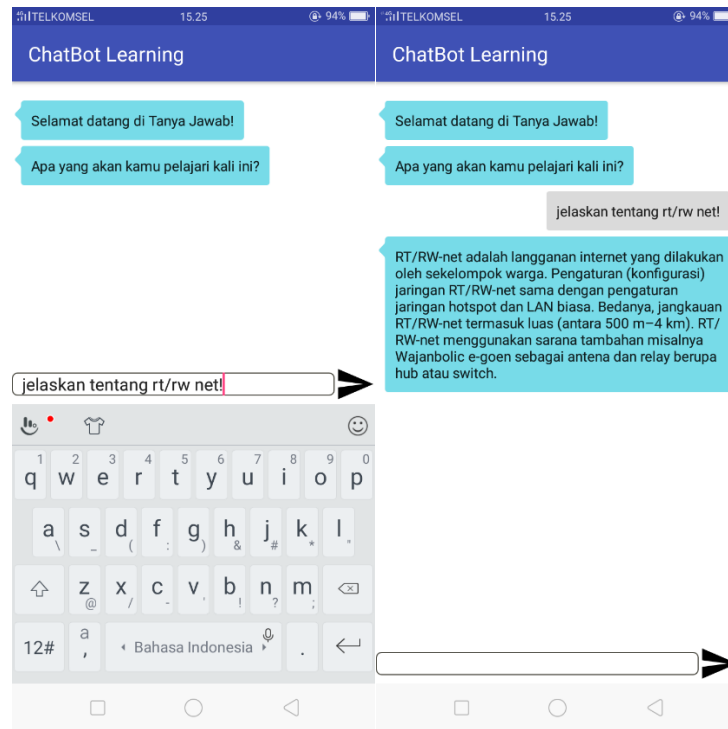
No	Nama Antarmuka	Keterangan
1	Halaman <i>Login</i>	Menampilkan form untuk memasukan username dan password.
2	Halaman <i>Utama</i>	Menampilkan fitur-fitur yang ada didalam aplikasi.
3	Halaman <i>Tanya Jawab</i>	Menampilkan form untuk memasukan pertanyaan yang ada didalam aplikasi.
4	Halaman <i>Baca Materi</i>	Menampilkan daftar dan materi yang ingin dibaca.
5	Halaman <i>Ganti Password</i>	Menampilkan form untuk mengganti password yang baru.
6	Halaman <i>Logout</i>	Menampilkan tombol logout untuk keluar dari aplikasi android.

4.1.4 Implementasi Teknologi

Implementasi teknologi merupakan penerapan dari pembahasan bab-bab sebelumnya kedalam bentuk nyata dari aplikasi atau sistem yang dibangun. Implementasi teknologi ini merupakan salah satu bagian yang sangat penting dalam penelitian dan pembangunan aplikasi *pembelajaran untuk chatbot* ini. Tanpa realisasi dari implementasi teknologi, sistem yang dibangun tidak dapat berfungsi sama sekali karena teknologi *mobile android dan web moodle* yang diterapkan merupakan salah satu integrasi sistem sehingga aplikasi dapat berjalan. Berikut adalah implementasi teknologi pada aplikasi pembelajaran untuk chatbot ini:

1. Chatbot

Sesuai dengan pembahasan pada bab-bab sebelumnya, teknologi Chatbot ini digunakan untuk memudahkan pengguna dalam memahami materi-materi pelajaran dengan tanya jawab seputar materi-materi pelajaran.



Gambar 4.1 Implementasi Teknologi Chatbot Pada Smartphone Android

Pada Gambar 4.7 Implementasi Teknologi Chatbot Pada Smartphone Android terlihat text area berwarna hitam lalu pengguna mengetik pertanyaan seputar materi-materi pelajaran setelah itu menekan simbol kirim hasilnya pengguna akan mendapatkan inti jawaban dari pertanyaan tersebut. Selanjutnya jika pengguna memanfaatkan fitur tanya jawab ini maka pengguna dapat memahami materi-materi pelajaran secara cepat dan mudah. Berikut adalah *Source Code* untuk implementasi teknologi *Chatbot* :

```
private void sendMessage(String message) {
    ChatMessage chatMessage = new ChatMessage(message, true,
false);
    mAdapter.add(chatMessage);

    final AsyncTask<String, Void, AIResponse> task = new
AsyncTask<String, Void, AIResponse>() {

        private AIError aiError;
```

```

        @Override
        protected AIResponse doInBackground(final String...
params) {
            final AIRequest request = new AIRequest();
            String query = params[0];
            String event = params[1];

            if (!TextUtils.isEmpty(query))
                request.setQuery(query);
            if (!TextUtils.isEmpty(event))
                request.setEvent(new AIEvent(event));
            final String contextString = params[2];
            RequestExtras requestExtras = null;
            if (!TextUtils.isEmpty(contextString)) {
                final List<AIContext> contexts =
Collections.singletonList(new AIContext(contextString));
                requestExtras = new RequestExtras(contexts,
null);
            }

            try {
                return aiDataService.request(request,
requestExtras);
            } catch (final AIServiceException e) {
                aiError = new AIError(e);
                return null;
            }
        }

        @Override
        protected void onPostExecute(final AIResponse
response) {
            if (response != null) {
                onResult(response);
            } else {
                onError(aiError);
            }
        }

```

```

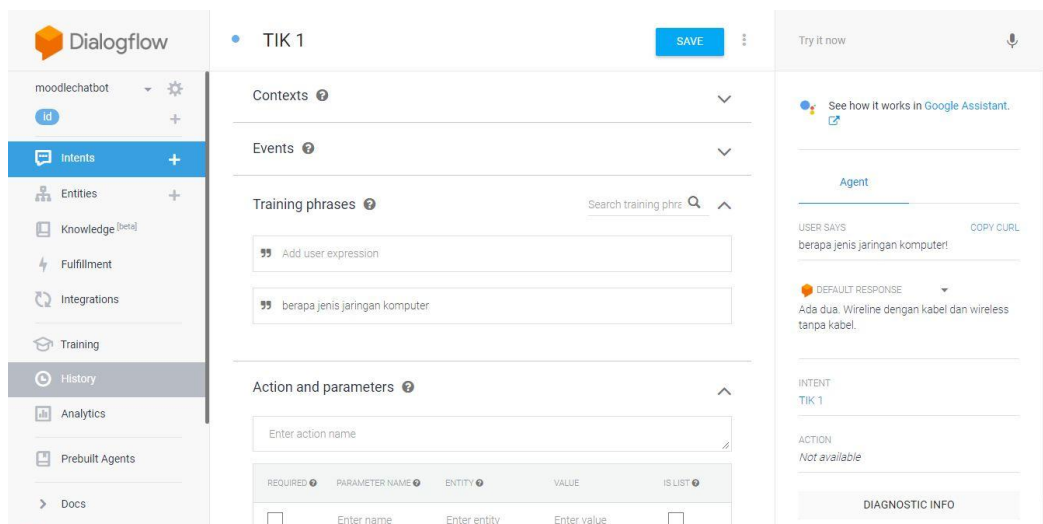
    }
};

task.execute(message, null, null);
}

```

2. API Dialogflow

Sesuai dengan pembahasan pada bab-bab sebelumnya, teknologi API Dialogflow ini digunakan untuk membangun aplikasi pembelajaran untuk chatbot. Dengan adanya API Dialogflow untuk pembuatan chatbot agar lebih memudahkan pengguna dalam proses tanya jawab seputar materi-materi pelajaran berdasarkan materi tanya jawab yang telah diupload pada *Backend*.



Gambar 4.2 Implementasi Teknologi API Dialogflow

Berikut adalah *Source Code* untuk implementasi teknologi *Android Call API Dialogflow*:

```

. . .
final AIConfiguration config = new
AIConfiguration("170a5a6b1b21473f8468d6573ba7f832",
                AIConfiguration.SupportedLanguages.DEFAULT,
                AIConfiguration.RecognitionEngine.System);

aiDataService = new AIDataService(this, config);

```

```

. . .
final AsyncTask<String, Void, AIResponse> task = new
AsyncTask<String, Void, AIResponse>() {

    private AIError aiError;

    @Override
    protected AIResponse doInBackground(final String...
params) {

        final AIRequest request = new AIRequest();
        String query = params[0];
        String event = params[1];

        if (!TextUtils.isEmpty(query))
            request.setQuery(query);
        if (!TextUtils.isEmpty(event))
            request.setEvent(new AIEvent(event));
        final String contextString = params[2];
        RequestExtras requestExtras = null;
        if (!TextUtils.isEmpty(contextString)) {
            final List<AIContext> contexts =
Collections.singletonList(new AIContext(contextString));
            requestExtras = new RequestExtras(contexts,
null);
        }

        try {
            return aiDataService.request(request,
requestExtras);
        } catch (final AIServiceException e) {
            aiError = new AIError(e);
            return null;
        }
    }

    @Override
    protected void onPostExecute(final AIResponse

```



```

response) {
    if (response != null) {
        onResult(response);
    } else {
        onError(aiError);
    }
}
};

task.execute(message, null, null);
. . .
private void onResult(final AIResponse response) {
    runOnUiThread(new Runnable() {
        @Override
        public void run() {
            Log.d(TAG, "onResult");

            final Result result = response.getResult();
            // resultTextView.setText(gson.toJson(response));

            receiveMessage(result.getFulfillment().getSpeech());
            . . .
        }

    });
}

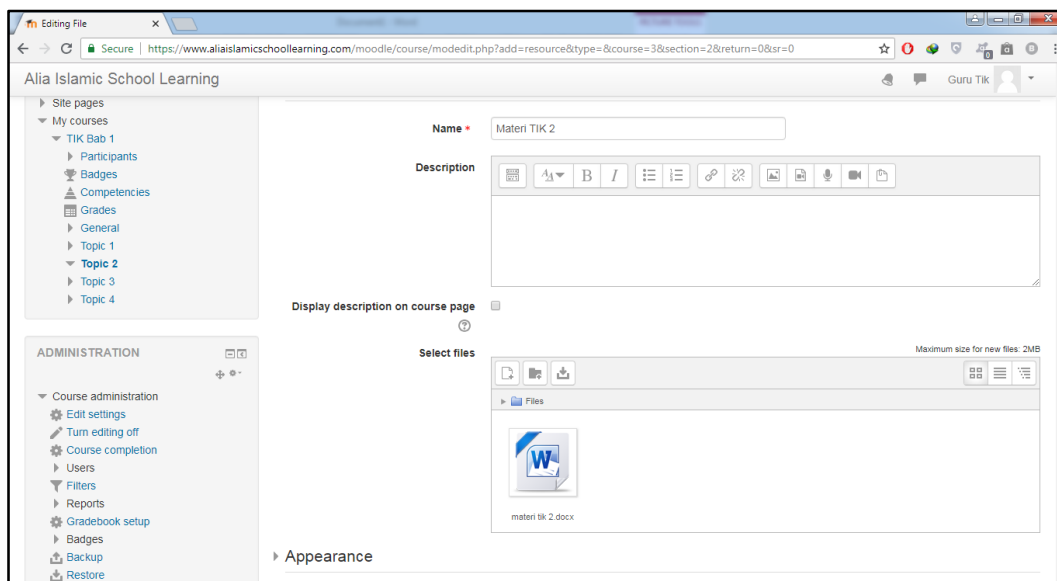
private void onError(final AIError error) {
    runOnUiThread(new Runnable() {
        @Override
        public void run() {
            //resultTextView.setText(error.toString());
            receiveMessage(error.toString());
        }

    });
}
}

```

3. Moodle

Sesuai dengan pembahasan pada bab-bab sebelumnya, teknologi Moodle (LMS) ini digunakan untuk membangun aplikasi pembelajaran untuk chatbot. Dengan adanya Moodle untuk pembuatan *Backend* lebih memudahkan pegawai dalam proses upload materi-materi tanya jawab yang nantinya akan terintegrasi dengan fitur tanya jawab yang ada di smartphone androidnya.



Gambar 4.3 Implementasi Teknologi Moodle (LMS)

Berikut adalah *Source Code* untuk implementasi teknologi *Moodle Call API*

Dialogflow:

```

echo '<script src="//code.jquery.com/jquery-
1.11.1.min.js"></script>
      <script type="text/javascript">
        function create(name,usersays,message) {
          var accessToken =
"96e30b84de614e24b40da6be70bdde55";

          $.ajax({
            type: "POST",
            url:
"https://api.dialogflow.com/v1/intents?v=20150910",
            contentType: "application/json;
charset=utf-8",
            //dataType: "json",
            headers: {
              "Authorization": "Bearer " +
accessToken
            },
            data: JSON.stringify({

```

```

        "name": name,
        "responses": [{"messages":
[{"speech": message, "type": 0}]}],
        "userSays": [{"data":
[{"text": usersays}]}]
    }},
    success: function(data) {
        val = JSON.stringify(data,
undefined, 2);
        console.log(val);
//document.getElementById("text").innerHTML=val;
    },
    error: function(data) {
        val = JSON.stringify(data,
undefined, 2);
        console.log(val);
//document.getElementById("text").innerHTML=val;
    }
    });
}
</script>';

for ($i=0;$i<count($parse['question']);$i++){
    $t=$parse['title'].' ' .($i+1);
    if($parse['answer'][$i]){
        echo '<script type="text/javascript">',
'create("'.$t.'", "'.$parse['question'][$i].'", "'.$parse['answer'
] [$i].'");',
        '</script>'
    ;
    }
}
}

```

4.1.5 Implementasi Basis Data

Implementasi basis data merupakan gambaran dalam pembuatan basis data pada aplikasi yang dibangun. Basis data yang digambarkan menggunakan Bahasa Pemrograman SQL (*Structured Query Language*). Implementasi basis data dapat dilihat pada table berikut :

Tabel 4.7 Implementasi Basis Data

No	Nama Tabel	SQL
----	------------	-----

1	<i>user</i>	<pre> CREATE TABLE `user` (`id` bigint(10) NOT NULL, `auth` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'manual', `confirmed` tinyint(1) NOT NULL DEFAULT '0', `policyagreed` tinyint(1) NOT NULL DEFAULT '0', `deleted` tinyint(1) NOT NULL DEFAULT '0', `suspended` tinyint(1) NOT NULL DEFAULT '0', `mnethostid` bigint(10) NOT NULL DEFAULT '0', `username` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `password` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `idnumber` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `firstname` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `lastname` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `email` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `emailstop` tinyint(1) NOT NULL DEFAULT '0', `icq` varchar(15) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `skype` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `yahoo` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `aim` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `msn` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `phone1` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `phone2` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `institution` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `department` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `address` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `city` varchar(120) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `country` varchar(2) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `lang` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'en', `calendartype` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'gregorian', `theme` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `timezone` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '99', `firstaccess` bigint(10) NOT NULL DEFAULT '0', </pre>
---	-------------	--

		<pre> `lastaccess` bigint(10) NOT NULL DEFAULT '0', `lastlogin` bigint(10) NOT NULL DEFAULT '0', `currentlogin` bigint(10) NOT NULL DEFAULT '0', `lastip` varchar(45) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `secret` varchar(15) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `picture` bigint(10) NOT NULL DEFAULT '0', `url` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `description` longtext COLLATE utf8mb4_unicode_ci, `descriptionformat` tinyint(2) NOT NULL DEFAULT '1', `mailformat` tinyint(1) NOT NULL DEFAULT '1', `maildigest` tinyint(1) NOT NULL DEFAULT '0', `maildisplay` tinyint(2) NOT NULL DEFAULT '2', `autosubscribe` tinyint(1) NOT NULL DEFAULT '1', `trackforums` tinyint(1) NOT NULL DEFAULT '0', `timecreated` bigint(10) NOT NULL DEFAULT '0', `timemodified` bigint(10) NOT NULL DEFAULT '0', `trustbitmask` bigint(10) NOT NULL DEFAULT '0', `imagealt` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `lastnamephonetic` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `firstnamephonetic` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `middlename` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `alternatename` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='One record for each person' ROW_FORMAT=COMPRESSED; ALTER TABLE `user` ADD PRIMARY KEY (`id`), ADD UNIQUE KEY `user_mneuse_uix` (`mneuse``,`username`), ADD KEY `user_del_ix` (`deleted`), ADD KEY `user_con_ix` (`confirmed`), ADD KEY `user_fir_ix` (`firstname`), ADD KEY `user_las_ix` (`lastname`), ADD KEY `user_cit_ix` (`city`), ADD KEY `user_cou_ix` (`country`), ADD KEY `user_las2_ix` (`lastaccess`), ADD KEY `user_ema_ix` (`email`), ADD KEY `user_aut_ix` (`auth`), ADD KEY `user_idn_ix` (`idnumber`), ADD KEY `user_fir2_ix` (`firstnamephonetic`), ADD KEY `user_las3_ix` (`lastnamephonetic`), ADD KEY `user_mid_ix` (`middlename`), </pre>
--	--	---

		<pre> ADD KEY `user_alt_ix` (`alternatename`); ALTER TABLE `user` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=1; </pre>
2	<i>course</i>	<pre> CREATE TABLE `course` (`id` bigint(10) NOT NULL, `category` bigint(10) NOT NULL DEFAULT '0', `sortorder` bigint(10) NOT NULL DEFAULT '0', `fullname` varchar(254) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `shortname` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `idnumber` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `summary` longtext COLLATE utf8mb4_unicode_ci, `summaryformat` tinyint(2) NOT NULL DEFAULT '0', `format` varchar(21) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'topics', `showgrades` tinyint(2) NOT NULL DEFAULT '1', `newsitems` mediumint(5) NOT NULL DEFAULT '1', `startdate` bigint(10) NOT NULL DEFAULT '0', `enddate` bigint(10) NOT NULL DEFAULT '0', `marker` bigint(10) NOT NULL DEFAULT '0', `maxbytes` bigint(10) NOT NULL DEFAULT '0', `legacyfiles` smallint(4) NOT NULL DEFAULT '0', `showreports` smallint(4) NOT NULL DEFAULT '0', `visible` tinyint(1) NOT NULL DEFAULT '1', `visibleold` tinyint(1) NOT NULL DEFAULT '1', `groupmode` smallint(4) NOT NULL DEFAULT '0', `groupmodeforce` smallint(4) NOT NULL DEFAULT '0', `defaultgroupingid` bigint(10) NOT NULL DEFAULT '0', `lang` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `calendartype` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `theme` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `timecreated` bigint(10) NOT NULL DEFAULT '0', `timemodified` bigint(10) NOT NULL DEFAULT '0', `requested` tinyint(1) NOT NULL DEFAULT '0', `enablecompletion` tinyint(1) NOT NULL DEFAULT '0', </pre>

		<pre> `completionnotify` tinyint(1) NOT NULL DEFAULT '0', `cacherev` bigint(10) NOT NULL DEFAULT '0') ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='Central course table' ROW_FORMAT=COMPRESSED; ALTER TABLE `course` ADD PRIMARY KEY (`id`), ADD KEY `cour_cat_ix` (`category`), ADD KEY `cour_idn_ix` (`idnumber`), ADD KEY `cour_sho_ix` (`shortname`), ADD KEY `cour_sor_ix` (`sortorder`); ALTER TABLE `course` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=1; </pre>
3	<i>Files</i>	<pre> CREATE TABLE `files` (`id` bigint(10) NOT NULL, `contenthash` varchar(40) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `pathnamehash` varchar(40) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `contextid` bigint(10) NOT NULL, `component` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `filearea` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `itemid` bigint(10) NOT NULL, `filepath` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `filename` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `userid` bigint(10) DEFAULT NULL, `filesize` bigint(10) NOT NULL, `mimetype` varchar(100) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `status` bigint(10) NOT NULL DEFAULT '0', `source` longtext COLLATE utf8mb4_unicode_ci, `author` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `license` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `timecreated` bigint(10) NOT NULL, `timemodified` bigint(10) NOT NULL, `sortorder` bigint(10) NOT NULL DEFAULT '0', `referencefileid` bigint(10) DEFAULT NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='description of files, content is stored in sha1 file pool' ROW_FORMAT=COMPRESSED; ALTER TABLE `files` ADD PRIMARY KEY (`id`), ADD UNIQUE KEY `file_pat_uix` (`pathnamehash`), ADD KEY `file_comfilconite_ix` (`component`,`filearea`,`contextid`,`itemid`), </pre>

		<pre> ADD KEY `file_con_ix` (`contenthash`), ADD KEY `file_con2_ix` (`contextid`), ADD KEY `file_use_ix` (`userid`), ADD KEY `file_ref_ix` (`referencefileid`); ALTER TABLE `files` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=1; </pre>
4	<i>role</i>	<pre> CREATE TABLE `role` (`id` bigint(10) NOT NULL, `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `shortname` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `description` longtext COLLATE utf8mb4_unicode_ci NOT NULL, `sortorder` bigint(10) NOT NULL DEFAULT '0', `archetype` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '') ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='moodle roles' ROW_FORMAT=COMPRESSED; INSERT INTO `role` (`id`, `name`, `shortname`, `description`, `sortorder`, `archetype`) VALUES (1, '', 'manager', '', 1, 'manager'), (2, '', 'coursecreator', '', 2, 'coursecreator'), (3, '', 'editingteacher', '', 3, 'editingteacher'), (4, '', 'teacher', '', 4, 'teacher'), (5, '', 'student', '', 5, 'student'), (6, '', 'guest', '', 6, 'guest'), (7, '', 'user', '', 7, 'user'), (8, '', 'frontpage', '', 8, 'frontpage'); ALTER TABLE `role` ADD PRIMARY KEY (`id`), ADD UNIQUE KEY `role_sor_uix` (`sortorder`), ADD UNIQUE KEY `role_sho_uix` (`shortname`); ALTER TABLE `role` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=9; </pre>
5	<i>course_categories</i>	<pre> CREATE TABLE `course_categories` (`id` bigint(10) NOT NULL, `name` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `idnumber` varchar(100) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `description` longtext COLLATE utf8mb4_unicode_ci, `descriptionformat` tinyint(2) NOT NULL DEFAULT '0', `parent` bigint(10) NOT NULL DEFAULT '0', `sortorder` bigint(10) NOT NULL DEFAULT '0', `coursecount` bigint(10) NOT NULL DEFAULT '0', `visible` tinyint(1) NOT NULL DEFAULT '1', </pre>

		<pre> `visibleold` tinyint(1) NOT NULL DEFAULT '1', `timemodified` bigint(10) NOT NULL DEFAULT '0', `depth` bigint(10) NOT NULL DEFAULT '0', `path` varchar(255) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `theme` varchar(50) COLLATE utf8mb4_unicode_ci DEFAULT NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='Course categories' ROW_FORMAT=COMPRESSED; ALTER TABLE `course_categories` ADD PRIMARY KEY (`id`), ADD KEY `courcate_par_ix` (`parent`); ALTER TABLE `course_categories` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=1; </pre>
6	<i>course_format_options</i>	<pre> CREATE TABLE `course_format_options` (`id` bigint(10) NOT NULL, `courseid` bigint(10) NOT NULL, `format` varchar(21) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `sectionid` bigint(10) NOT NULL DEFAULT '0', `name` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `value` longtext COLLATE utf8mb4_unicode_ci) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='Stores format-specific options for the course or course sect' ROW_FORMAT=COMPRESSED; INSERT INTO `course_format_options` (`id`, `courseid`, `format`, `sectionid`, `name`, `value`) VALUES (1, 1, 'site', 0, 'numsections', '1'), (2, 2, 'topics', 0, 'hiddensections', '0'), (3, 2, 'topics', 0, 'coursedisplay', '0'), (4, 3, 'topics', 0, 'hiddensections', '0'), (5, 3, 'topics', 0, 'coursedisplay', '0'); ALTER TABLE `course_format_options` ADD PRIMARY KEY (`id`), ADD UNIQUE KEY `courformopti_couforsecnam_uix` (`courseid`, `format`, `sectionid`, `name`), ADD KEY `courformopti_cou_ix` (`courseid`); ALTER TABLE `course_format_options` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6; </pre>

7	<i>enrol</i>	<pre> CREATE TABLE `enrol` (`id` bigint(10) NOT NULL, `enrol` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '', `status` bigint(10) NOT NULL DEFAULT '0', `courseid` bigint(10) NOT NULL, `sortorder` bigint(10) NOT NULL DEFAULT '0', `name` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `enrolperiod` bigint(10) DEFAULT '0', `enrolstartdate` bigint(10) DEFAULT '0', `enrolenddate` bigint(10) DEFAULT '0', `expirynotify` tinyint(1) DEFAULT '0', `expirythreshold` bigint(10) DEFAULT '0', `notifyall` tinyint(1) DEFAULT '0', `password` varchar(50) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `cost` varchar(20) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `currency` varchar(3) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `roleid` bigint(10) DEFAULT '0', `customint1` bigint(10) DEFAULT NULL, `customint2` bigint(10) DEFAULT NULL, `customint3` bigint(10) DEFAULT NULL, `customint4` bigint(10) DEFAULT NULL, `customint5` bigint(10) DEFAULT NULL, `customint6` bigint(10) DEFAULT NULL, `customint7` bigint(10) DEFAULT NULL, `customint8` bigint(10) DEFAULT NULL, `customchar1` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `customchar2` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `customchar3` varchar(1333) COLLATE utf8mb4_unicode_ci DEFAULT NULL, `customdec1` decimal(12,7) DEFAULT NULL, `customdec2` decimal(12,7) DEFAULT NULL, `customtext1` longtext COLLATE utf8mb4_unicode_ci, `customtext2` longtext COLLATE utf8mb4_unicode_ci, `customtext3` longtext COLLATE utf8mb4_unicode_ci, `customtext4` longtext COLLATE utf8mb4_unicode_ci, `timecreated` bigint(10) NOT NULL DEFAULT '0', `timemodified` bigint(10) NOT NULL DEFAULT '0') ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci COMMENT='Instances of enrolment plugins used in courses, fields marke' ROW_FORMAT=COMPRESSED; ALTER TABLE `enrol` ADD PRIMARY KEY (`id`), ADD KEY `enro_enr_ix` (`enrol`), ADD KEY `enro_cou_ix` (`courseid`); </pre>
---	--------------	---

		<pre>ALTER TABLE `enrol` MODIFY `id` bigint(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=1;</pre>
--	--	--

4.2. Pengujian Sistem

Pengujian merupakan bagian yang penting dalam siklus pengembangan perangkat lunak. Tujuan dari pengujian ini adalah untuk menjamin bahwa perangkat lunak yang dibangun memiliki kualitas yang bagus. Pengujian terhadap program itu sendiri bertujuan agar program dapat berjalan dengan baik tanpa mengalami kerusakan sistem atau gangguan, dan memungkinkan untuk dilakukannya pengembangan sistem lebih lanjut. Pengujian perangkat lunak ini menggunakan metode pengujian blackbox. Pengujian blackbox ini tidak perlu tahu apa yang sesungguhnya terjadi dalam sistem atau perangkat lunak, yang diuji adalah masukan serta keluarannya. Berikut ini adalah rencana pengujian Pembangunan Aplikasi Pembelajaran Untuk Chatbot ini.

4.2.1. Pengujian Alpha

Pengujian *alpha* yang dilakukan adalah menggunakan metode *BlackBox* yang berfokus pada titik fungsionalitas dari sistem yang dibangun.

4.2.1.1. Pengujian Blackbox

Pengujian *BlackBox* merupakan pengujian fungsional yang diadakan dilingkungan pembangunan oleh sekumpulan pengguna yang akan menggunakan sistem yang dibangun. Pihak pembangun melihat serta mencatat kesalahan dan permasalahan yang dirasakan oleh pengguna. Pengujian *BlackBox* yang dilakukan pada sistem aplikasi *backend* dan sistem aplikasi *frontend* yang dibangun.

4.2.1.2.Skenario Pengujian

Skenario pengujian memaparkan urutan dan hal yang diuji pada aplikasi *backend* dan aplikasi *frontend*. Adapun skenario pengujian yang dilakukan adalah sebagai berikut ini :

1. Skenario Pengujian Aplikasi *Backend*

Skenario pengujian aplikasi *backend* memaparkan urutan dan hal yang diuji pada aplikasi *backend* Pembangunan Aplikasi Pembelajaran Untuk Chatbot berbasis Android. Skenario pengujian aplikasi *backend* dapat dilihat pada Tabel 4.8.

Tabel 4.8 Skenario Pengujian Aplikasi Backend

No	Komponen yang Diuji	Skenario	Pengujian
1	Halaman <i>Login</i>	Input form untuk memasukan username dan password benar.	BlackBox
		Input form untuk memasukan username dan password salah.	Blackbox
2	Halaman <i>Menambah Pengguna</i>	Input form untuk menambah data pengguna yang baru.	BlackBox
3	Halaman <i>Mengubah Pengguna</i>	Input form dari data pengguna yang akan diubah.	BlackBox
4	Halaman <i>Menghapus Pengguna</i>	Menghapus data pengguna yang akan dihapus.	BlackBox
5	Halaman <i>Menambah Kategori Kursus</i>	Input form untuk menambah data kategori kursus yang baru.	BlackBox
6	Halaman <i>Mengubah Kategori Kursus</i>	Input form dari data kategori kursus yang akan diubah.	BlackBox
7	Halaman <i>Menghapus Kategori Kursus</i>	Menghapus data kategori kursus yang akan dihapus.	BlackBox
8	Halaman <i>Menambah Kursus</i>	Input form untuk menambah data kursus yang baru.	BlackBox
9	Halaman <i>Mengubah Kursus</i>	Input form dari data kursus yang akan diubah.	BlackBox
10	Halaman <i>Menghapus Kursus</i>	Menghapus data kursus yang akan dihapus.	BlackBox
11	Halaman <i>Upload Materi</i>	Input form untuk mengupload data materi yang baru.	BlackBox

12	Halaman <i>Logout</i>	Menekan tombol logout untuk keluar dari website.	BlackBox
----	-----------------------	--	----------

2. Skenario Pengujian Aplikasi *Frontend*

Skenario pengujian aplikasi *frontend* memaparkan urutan dan hal yang diuji pada aplikasi *frontend* Pembangunan Aplikasi Pembelajaran Untuk Chatbot berbasis Android. Skenario pengujian aplikasi *frontend* dapat dilihat pada Tabel 4.8.

Tabel 4.9 Skenario Pengujian Aplikasi Frontend

No	Komponen yang Diuji	Skenario	Pengujian
1	Halaman <i>Login</i>	Input form untuk memasukan username dan password benar.	BlackBox
		Input form untuk memasukan username dan password salah.	BlackBox
2	Halaman <i>Tanya Jawab</i>	Input form untuk memasukan pertanyaan yang ada didalam aplikasi benar.	BlackBox
		Input data kosong.	Blackbox
3	Halaman <i>Baca Materi</i>	Melihat daftar dan materi yang ingin dibaca.	BlackBox
		Input data kosong	Blackbox
4	Halaman <i>Ganti Password</i>	Input form untuk mengganti password yang benar.	BlackBox
		Input form untuk mengganti password yang salah.	BlackBox
5	Halaman <i>Logout</i>	Menampilkan tombol logout untuk keluar dari aplikasi android.	BlackBox

4.2.1.3 Kasus dan Hasil Pengujian

Berdasarkan hasil pengujian data yang telah dibuat, maka bisa dilakukan pengujian sebagai berikut :

4.2.1.3.1 Hasil Pengujian Fungsionalitas Aplikasi Backend

Tabel 4.10 Hasil Pengujian Aplikasi Backend

No	Komponen yang Diuji	Skenario Pengujian	Hasil yang diharapkan	Kesimpulan
----	---------------------	--------------------	-----------------------	------------

1	Halaman <i>Login</i>	Input form untuk memasukkan username dan password benar.	Masuk ke halaman utama sesuai dengan status pegawainya	(√) Berhasil () Gagal
		Input form untuk memasukkan username dan password salah.	Muncul pesan Username dan password salah.	() Berhasil (√) Gagal
2	Halaman <i>Menambah Pengguna</i>	Input form untuk menambah data pengguna yang baru.	Menampilkan pesan data pengguna telah tersimpan.	(√) Berhasil () Gagal
3	Halaman <i>Mengubah Pengguna</i>	Input form dari data pengguna yang akan diubah.	Menampilkan pesan data pengguna telah berubah.	(√) Berhasil () Gagal
4	Halaman <i>Menghapus Pengguna</i>	Menghapus data pengguna yang akan dihapus.	Menampilkan pesan data pengguna telah terhapus.	(√) Berhasil () Gagal
5	Halaman <i>Menambah Kategori Kursus</i>	Input form untuk menambah data kategori kursus yang baru.	Menampilkan pesan data kategori kursus telah tersimpan.	(√) Berhasil () Gagal
6	Halaman <i>Mengubah Kategori Kursus</i>	Input form dari data kategori kursus yang akan diubah.	Menampilkan pesan data kategori kursus telah berubah.	(√) Berhasil () Gagal
7	Halaman <i>Menghapus Kategori Kursus</i>	Menghapus data kategori kursus yang akan dihapus.	Menampilkan pesan data kategori kursus telah terhapus.	(√) Berhasil () Gagal
8	Halaman <i>Menambah Kursus</i>	Input form untuk menambah data kursus yang baru.	Menampilkan pesan data kursus telah tersimpan.	(√) Berhasil () Gagal
9	Halaman <i>Mengubah Kursus</i>	Input form dari data kursus yang akan diubah.	Menampilkan pesan data kursus telah berubah.	(√) Berhasil () Gagal
10	Halaman <i>Menghapus Kursus</i>	Menghapus data kursus yang akan dihapus.	Menampilkan pesan data kursus telah terhapus.	(√) Berhasil () Gagal
11	Halaman <i>Upload Materi</i>	Input form untuk mengupload data materi yang baru.	Menampilkan pesan data materi telah tersimpan.	(√) Berhasil () Gagal
12	Halaman <i>Logout</i>	Menekan tombol logout untuk keluar dari website.	Kembali ke halaman login.	(√) Berhasil () Gagal

4.2.1.3.2 Hasil Pengujian Fungsionalitas Aplikasi Frontend

Tabel 4.11 Hasil Pengujian Aplikasi Frontend

No	Komponen yang Diuji	Skenario	Hasil yang diharapkan	Kesimpulan
1	Halaman <i>Login</i>	Input form untuk memasukkan username dan password benar.	Masuk ke halaman utama sesuai dengan status siswa.	(√) Berhasil () Gagal
		Input form untuk memasukkan username dan password salah.	Muncul pesan Username dan password salah.	() Berhasil (√) Gagal
2	Halaman <i>Tanya Jawab</i>	Input form untuk memasukkan pertanyaan yang ada didalam aplikasi benar.	Menampilkan jawaban dari pesan data yang diinputkan.	(√) Berhasil () Gagal
		Input data kosong.	Input form tidak diproses.	() Berhasil (√) Gagal
3	Halaman <i>Baca Materi</i>	Melihat daftar dan materi yang ingin dibaca.	Menampilkan materi dari data yang diinputkan.	(√) Berhasil () Gagal
		Input data kosong	Input form tidak diproses.	() Berhasil (√) Gagal
4	Halaman <i>Ganti Password</i>	Input form untuk mengganti password yang benar.	Menampilkan pesan data password telah terubah.	(√) Berhasil () Gagal
		Input form untuk mengganti password yang salah.	Menampilkan pesan gagal ganti password karena kesalahan input data.	() Berhasil (√) Gagal
5	Halaman <i>Logout</i>	Menampilkan tombol logout untuk keluar dari aplikasi android.	Kembali ke halaman login.	(√) Berhasil () Gagal

4.2.2 Kesimpulan Pengujian BlackBox

Berdasarkan hasil pengujian BlackBox yang dilakukan dengan kasus uji fungsionalitas diatas, dapat ditarik kesimpulan sebagai berikut:

1. Jika data yang dimasukan valid, maka sistem akan mengeluarkan informasi sesuai dengan harapan yang diinginkan.
2. Jika data yang dimasukan tidak valid dengan sistem, maka sistem akan menunjukkan pesan kesalahan sesuai dengan kesalahannya. Sehingga disaat

memasukan data harus yang sesuai dengan aturan yang terdapat pada sistemnya, sehingga fungsional sistem dapat mengeluarkan hasil yang sesuai.

4.2.3 Pengujian Beta

Pengujian beta yang mana merupakan bagian dari acceptance testing atau User Acceptance Testing (UAT) merupakan pengujian yang dilakukan secara objektif yang dilakukan secara langsung oleh pengguna aplikasi. Pengujian beta dilakukan menggunakan teknik pengolahan data yaitu kuesioner. Berikut ini adalah pertanyaan yang diajukan kepada siswa :

Tabel 4.12 Tabel Pertanyaan Kuesioner

No	Pertanyaan	SS	S	KS	TS	STS
1	Apakah aplikasi Pembelajaran Untuk Chatbot ini dapat mempermudah anda untuk mendapatkan materi-materi secara cepat dan mudah ketika berada diluar jam sekolah ?					
2	Apakah aplikasi Pembelajaran Untuk Chatbot ini dapat mempermudah anda untuk memahami tentang materi-materi ketika izin tidak masuk sekolah ?					
3	Apakah aplikasi Pembelajaran Untuk Chatbot ini mudah untuk dipelajari dan digunakan anda ?					
4	Apakah aplikasi Pembelajaran Untuk Chatbot ini dapat mempermudah anda dalam mendapatkan inti jawaban dari materi-materi yang ada di sekolah ?					

4.2.3.1 Skenario Pengujian Beta Dengan Data Kuesioner

Kuesioner pengujian merupakan teknik pengolahan data yang merupakan langkah dalam mengolah data yang telah didapat untuk dijadikan hasil penelitian sehingga dapat ditarik kesimpulan. Kuesioner yang akan di ajukan kepada responden mempunyai pertanyaan dengan dengan pilihan jawaban 5 buah menggunakan skala likert 1 sampai 5. Berikut adalah detail skala likert:

Tabel 4.13 Tabel Penilaian Skala Likert

Kategori	Nilai
Sangat Setuju	5
Setuju	4
Kurang Setuju	3

Tidak Setuju	2
Sangat Tidak Setuju	1

1. Skor ideal merupakan skor yang digunakan untuk menghitung skor untuk menentukan rating scale dan jumlah seluruh jawaban. Untuk menghitung jumlah skor ideal (kriterium) dari seluruh item, digunakan rumus berikut yaitu:

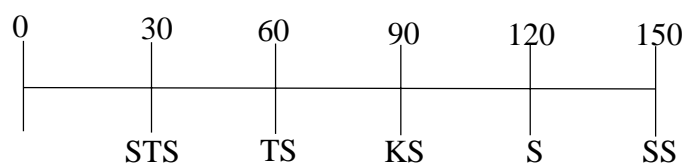
$$\text{Skor Ideal} = \text{Nilai skala} \times \text{Jumlah responden}$$

Skor tertinggi adalah 5 dan jumlah responden 30, maka dapat dirumuskan menjadi:

Tabel 4.14 Tabel Perhitungan Skala

Rumus	Skala
$5 \times 30 = 150$	Sangat Setuju
$4 \times 30 = 120$	Setuju
$3 \times 30 = 90$	Kurang Setuju
$2 \times 30 = 60$	Tidak Setuju
$1 \times 30 = 30$	Sangat Tidak Setuju

2. Selanjutnya semua jawaban responden dijumlahkan dan dimasukkan kedalam rating scale dan ditentukan daerah jawabannya.



Gambar 4.4 Rating Scale

Rating scale berfungsi untuk mengetahui hasil data kuisisioner secara umum dan keseluruhan yang didapat dari penilaian angket kuisisioner. Dengan ketentuan sebagai berikut ini:

Tabel 4.15 Tabel Nilai Skala

Nilai Jawaban	Skala
121 - 150	Sangat Setuju
91 – 120	Setuju
61 – 90	Kurang Setuju
31 - 60	Tidak Setuju
0 - 30	Sangat Tidak Setuju

3. Sedangkan untuk mengetahui jumlah jawaban dari para responden melalui persentase, yaitu digunakan rumus sebagai berikut ini:

$$p = \frac{f}{n} \times 100\%$$

Keterangan

p : Persentase

f : Frekuensi dari setiap jawaban
kuesioner

n : Jumlah skor ideal

100: Bilangan tetap

4.2.3.2 Perhitungan Hasil Kuesioner

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung nilainya dengan menggunakan rumur diatas. Kuesioner ini telah diujikan kepada 30 orang responden yang merupakan pengguna smartphone android. Hasil perhitungan persentase masing-masing jawaban dapat dilihat pada tabel berikut:

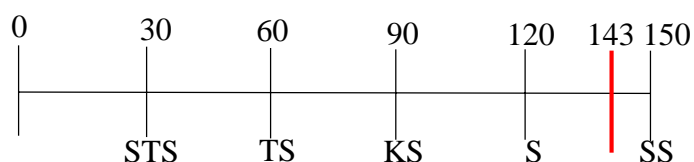
1. Apakah aplikasi Pembelajaran Untuk Chatbot ini dapat mempermudah anda untuk mendapatkan materi-materi secara cepat dan mudah ketika berada diluar jam sekolah ?

Tabel 4.16 Tabel Perhitungan Hasil Kuesioner Pertanyaan 1

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	26	130	(143 / (30x5))	
Setuju	4	1	4		

Kurang Setuju	3	3	9	x 100% = 95.3%	Sangat Setuju
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	143		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 143 skor dengan skala kategori jawaban diantaranya sangat setuju ,setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 95.3% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk memudahkan pengguna dalam mendapatkan materi-materi secara cepat dan mudah. Berikut gambar *interval rating scale* untuk pertanyaan nomer 1:



Gambar 4.5 Rating Scale Pertanyaan 1

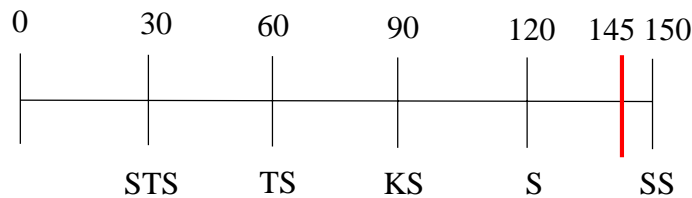
2. Apakah aplikasi Pembelajaran Untuk Chatbot ini dapat mempermudah anda untuk memahami tentang materi-materi ketika izin tidak masuk sekolah ?

Tabel 4.17 Tabel Perhitungan Hasil Kuesioner Pertanyaan 2

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	27	135	(145 / (30x5)) x 100% = 97%	Sangat Setuju
Setuju	4	1	4		
Kurang Setuju	3	2	6		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	145		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 145 skor dengan skala kategori jawaban diantaranya sangat setuju ,setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 97% dari nilai yang

diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk memudahkan pengguna dalam memahami tentang materi-materi. Berikut gambar *interval rating scale* untuk pertanyaan nomer 2:



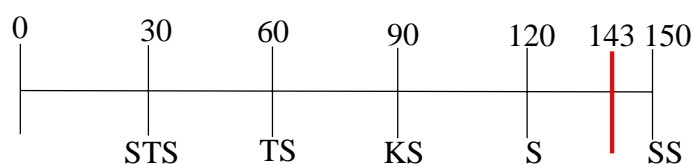
Gambar 4.6 Rating Scale Pertanyaan 2

3. Apakah aplikasi Pembelajaran Untuk Chatbot ini mudah untuk dipelajari dan digunakan anda ?

Tabel 4.18 Tabel Perhitungan Hasil Kuesioner Pertanyaan 3

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	26	130	$(143 / (30 \times 5))$ $\times 100\% =$ 95.3%	Sangat Setuju
Setuju	4	1	4		
Kurang Setuju	3	3	9		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	143		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 143 skor dengan skala kategori jawaban diantaranya sangat setuju ,setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 95.3% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk memudahkan pengguna untuk dipelajari dan digunakan. Berikut gambar *interval rating scale* untuk pertanyaan nomer 3:



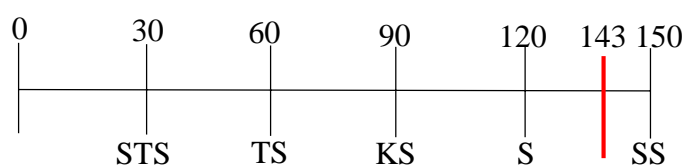
Gambar 4.7 Rating Scale Pertanyaan 3

4. Apakah aplikasi Pembelajaran Untuk Chatbot ini dapat mempermudah anda dalam mendapatkan inti jawaban dari materi-materi yang ada di sekolah ?

Tabel 4.19 Tabel Perhitungan Hasil Kuesioner Pertanyaan 4

Kategori Jawaban	Skor	Frekuensi Jawaban	Total Skor	Nilai Presentase	Keputusan
Sangat Setuju	5	26	130	(143 / (30x5)) x 100% = 95.3%	Sangat Setuju
Setuju	4	1	4		
Kurang Setuju	3	3	9		
Tidak Setuju	2	0	0		
Sangat Tidak Setuju	1	0	0		
Jumlah		30	143		

Berdasarkan perhitungan diatas, total skor yang didapat adalah sebanyak 143 skor dengan skala kategori jawaban diantaranya sangat setuju ,setuju dan kurang setuju. Sedangkan hasil dari nilai presentasi responden adalah 95.3% dari nilai yang diharapkan sebesar 100%. Maka dapat disimpulkan bahwa aplikasi ini sudah dapat memenuhi tujuannya untuk memudahkan pengguna mendapatkan inti jawaban dari materi-materi. Berikut gambar *interval rating scale* untuk pertanyaan nomer 4:



Gambar 4.8 Rating Scale Pertanyaan 4

4.2.4 Kesimpulan Pengujian Beta

Berdasarkan hasil persentase perhitungan pengujian beta kepada responden yang merupakan pengguna smartphone android, maka dapat disimpulkan bahwa Pembangunan Aplikasi Pembelajaran Untuk Chatbot Berbasis Android di SMA IT ALIA TANGERANG ini sudah sesuai dengan tujuan yang diharapkan oleh penulis.

