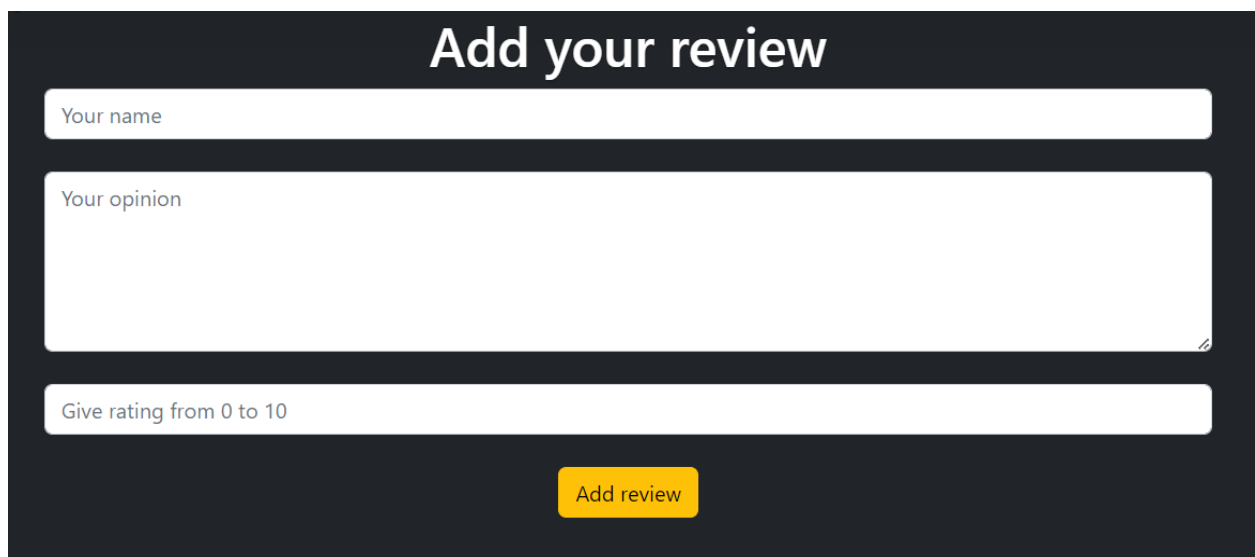


## Project report 10

For the last report we prepared some new stuff that we will add to the midterm 2 report. There is a website that has a main page, filter on main page that filters products, a review page, an adding item page, authorisation and registration page. Also, we improved and updated the design of all pages.

And finally, the last function is the function of commenting and giving a rate of the products, we call it review. Here we can text the name of user, feedback and rate (0-10)















The image shows a web form titled "Add your review" on a dark background. It contains three input fields: "Your name" (a single-line text box), "Your opinion" (a multi-line text area), and "Give rating from 0 to 10" (a single-line text box). Below these fields is a yellow button labeled "Add review".

Here are the reviews themselves that is already in database.

Reviews	
<div><b>Samat</b> jsdnjsadjnsadjsdna 8/10</div>	<div><b>ijdfjdf</b> djfndjnfjdjnf 9/10</div>
<div><b>Abdussamad</b> project bazar jok eken, tema 10/10</div>	<div><b>Tulpar</b> bazar jok 8/10</div>
<div><b>Abdussamad</b> wqeqwfsdfsdfsfsd 9/10</div>	<div><b>Baital</b> qwertyuiodjnfjdnfds 6/10</div>

+ Options

				id	name	text	rating
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	Samat	jsdnjsadjnsadjsdna	8
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	ijdfjdf	djfndjnfjdjnf	9
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	Abdussamad	project bazar jok eken, tema	10
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	Tulpar	bazar jok	8
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	Abdussamad	wqeqwfsdfsdfsfsd	9

The database table of the review.

## The code of the review function:

```
{{ define "review" }}

{{ template "header" }}

<main class="px-3">
  <h1 class="cover-heading">Add your review</h1>
  <form action="/save_review/" method="post">
    <input type="text" name="name" id="name" placeholder="Your name" class="form-control input"><br>
    <textarea name="text" id="text" cols="30" rows="5" class="form-control" placeholder="Your opinion"></textarea><br>
    <input type="number" name="rating" id="rating" placeholder="Give rating from 0 to 10" class="form-control input"><br>
    <button class="btn btn-warning">Add review</button>
  </form>

  <h1 class="cover-heading" style="padding-top: 100px;">Reviews</h1>
  <div style="display: flex; flex-wrap: wrap; width: 60%; margin: 0 auto; padding-bottom: 50px;">
    {{ range . }}
    <div class="alert alert-light" style="width: calc(1/2*100% - 10px);
margin: 5px; text-align: left;">
      <h2>{{ .Name }}</h2>
      <p>{{ .Text }}</p>
      <p>{{ .Rating }}/10</p>
    </div>
    {{ end }}
  </div>
</main>

{{ template "footer" }}

{{ end }}
```

```

func review(w http.ResponseWriter, r *http.Request){
    t, err := template.ParseFiles("templates/review.html", "templates/header.html", "templates/footer.html")

    if err != nil{
        fmt.Fprint(w, err.Error())
    }

    db, err := sql.Open("mysql", "root:root@tcp(127.0.0.1:3306)/golang")
    if err != nil {
        panic(err)
    }
    defer db.Close()

    res, err := db.Query("SELECT * FROM `review`")
    if err != nil {
        panic(err)
    }

    comments = []Review{}
    for res.Next(){
        var comment Review
        err = res.Scan(&comment.Id, &comment.Name, &comment.Text, &comment.Rating)
        if err != nil {
            panic(err)
        }

        comments = append(comments, comment)
    }
}

```

```

    t.ExecuteTemplate(w, "review", comments)
}

func save_review(w http.ResponseWriter, r *http.Request){
    name := r.FormValue("name")
    text := r.FormValue("text")
    rating, err := strconv.Atoi(r.FormValue("rating"))

    if err != nil {
        fmt.Println("Error during conversion")
    }

    if name == "" || text == ""{
        fmt.Fprintf(w, "Not all info fill in")
    } else{
        db, err := sql.Open("mysql", "root:root@tcp(127.0.0.1:3306)/golang")
        if err != nil {
            panic(err)
        }
        defer db.Close()

        insert, err := db.Query(fmt.Sprintf("INSERT INTO `review` (`name`,`text`,`rating`) VALUES('%s','%s','%d')", name, text, rating))
        if err != nil{
            panic(err)
        }
        defer insert.Close()
    }
}

```

```
    }  
    defer insert.Close()  
  
    http.Redirect(w, r, "/review/", http.StatusSeeOther)  
}  
}
```