App.py file:

Code before debugging:

```
from flask import Flask, render_template, request

app = Flask(__name__)

notes = []
@app.route('/', methods=["POST"])
def index():
    note = request.args.get("note")
    notes.append(note)
    return render_template("home.html", notes=notes)


if __name__ == '__main__':
    app.run(debug=True)
```

Code after debugging:

```
from flask import Flask, render_template, request, session
from flask_session import Session

app = Flask(__name__)
app.config["SESSION_PERMANENT"] = False
app.config["SESSION_TYPE"] = "filesystem"
Session(app)

notes = []
@app.route('/', methods=["GET","POST"])
def index():
    if session.get("notes") is None:
        session["notes"] = []
    if request.method =='POST':
        note = request.form.get("note")
        if note != "":
            session["notes"].append(note)

    return render_template("home.html", notes=session['notes'])

if __name__ == '__main__':
    app.run(debug=True)
```

Explanation:

1. First, I added “GET” and “POST” both the methods to the default route (‘/’) so that the home.html will get displayed when the application runs using GET method and the notes can get fetched using POST method as this method provide security to data from getting revealed.
2. Then the information fetched using POST also has to be converted by fetching the method from “.args” which is used for GET method to “.form” which is for POST method.
3. To make the application used by many users simultaneously, Flask-Session is imported so that every user can find their data individually. Sessions allow the user to get a specific value in the duration of their login and logout in any application.
4. *session.get(“notes”) is None* is added to check whether the user already have a list if not then provided with a new one i.e. an empty list.
5. *note !=* “” is added so that no empty string as a note.
6. Finally, the final list is passed to .html file by passing session[‘notes’].

home.html:

Code before debugging:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <form action="">
    <input type="text" name="note" placeholder="Enter a note">
    <button>Add Note</button>
  </form>

  <ul>
  {% for note in notes%}
    <li>{{ note }}</li>
  {% endfor %}
  </ul>
</body>
</html>
```

Code after debugging:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Note Taking App</title>
</head>
```

```
<body>
  <form action="" method="post">
    <input type="text" name="note" placeholder="Enter a note">
    <button>Add a Note</button>
  </form>

  <ul>
  {% for note in notes %}
    <li>{{ note }}</li>
  {% endfor %}
  </ul>
</body>
</html>
```

Explanation:

1. Here I have just have to add a POST method so that the values fetched through the form is sent back to POST method in the backend file.