

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

Python Files

File: app.py

```
1 | from flask import Flask, render_template, request, redirect, session
2 | import sqlite3, os
3 | from datetime import datetime
4 |
5 | # ===== APP =====
6 | app = Flask(__name__)
7 | app.secret_key = "kespo_secret_key"
8 |
9 | BASE_DIR = os.path.dirname(os.path.abspath(__file__))
10 | DB = os.path.join(BASE_DIR, "kespo.db")
11 | UPLOADS = os.path.join(BASE_DIR, "static", "uploads")
12 | os.makedirs(UPLOADS, exist_ok=True)
13 |
14 | # ===== DB =====
15 | def db():
16 |     c = sqlite3.connect(DB)
17 |     c.row_factory = sqlite3.Row
18 |     return c
19 |
20 | def init_db():
21 |     c = db()
22 |     cur = c.cursor()
23 |
24 |     cur.execute("""
25 |         CREATE TABLE IF NOT EXISTS farmer(
26 |             id INTEGER PRIMARY KEY AUTOINCREMENT,
27 |             name TEXT,
28 |             email TEXT UNIQUE,
29 |             phone TEXT,
30 |             password TEXT,
31 |             trust_tier TEXT DEFAULT 'basic'
32 |         )
33 |         """
34 |     )
35 |
36 |     cur.execute("""
37 |         CREATE TABLE IF NOT EXISTS harvest(
38 |             id INTEGER PRIMARY KEY AUTOINCREMENT,
39 |             farmer_id INTEGER,
40 |             quantity TEXT,
41 |             expected_price TEXT,
42 |             image TEXT,
43 |             harvest_date TEXT,
44 |             status TEXT DEFAULT 'pending'
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
44 |     )
45 |     """ )
46 |
47 |     cur.execute("""
48 |         CREATE TABLE IF NOT EXISTS deal(
49 |             id INTEGER PRIMARY KEY AUTOINCREMENT,
50 |             harvest_id INTEGER,
51 |             buyer_name TEXT,
52 |             buyer_email TEXT,
53 |             deal_value REAL,
54 |             commission REAL,
55 |             farmer_net REAL,
56 |             status TEXT DEFAULT 'initiated',
57 |             created_at TEXT
58 |         )
59 |     """ )
60 |
61 |     c.commit()
62 |     c.close()
63 |
64 | # ====== ROOT ======
65 | @app.route("/")
66 | def root():
67 |     return redirect("/login")
68 |
69 | # ====== AUTH ======
70 | @app.route("/login", methods=["GET", "POST"])
71 | def login():
72 |     if request.method == "POST":
73 |         c = db(); cur = c.cursor()
74 |         cur.execute(
75 |             "SELECT * FROM farmer WHERE email=? AND password=?",
76 |             (request.form["email"], request.form["password"]))
77 |         )
78 |         f = cur.fetchone()
79 |         c.close()
80 |         if f:
81 |             session.clear()
82 |             session["farmer_id"] = f["id"]
83 |             session["farmer_name"] = f["name"]
84 |             return redirect("/dashboard")
85 |     return render_template("login.html")
86 |
87 | @app.route("/logout")
88 | def logout():
89 |     session.clear()
90 |     return redirect("/login")
91 |
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
92 | # ===== FARMER =====
93 | @app.route("/dashboard")
94 | def dashboard():
95 |     if "farmer_id" not in session:
96 |         return redirect("/login")
97 |     c = db(); cur = c.cursor()
98 |     cur.execute("SELECT trust_tier FROM farmer WHERE id=?", (session["farmer_id"],))
99 |     tier = cur.fetchone()["trust_tier"]
100 |    cur.execute("SELECT COUNT(*) FROM harvest WHERE farmer_id=?", (session["farmer_id"],))
101 |    count = cur.fetchone()[0]
102 |    c.close()
103 |    return render_template("dashboard.html",
104 |                           farmer_name=session["farmer_name"],
105 |                           trust_tier=tier,
106 |                           harvest_count=count
107 | )
108 |
109 | @app.route("/upload", methods=["GET", "POST"])
110 | def upload():
111 |     if "farmer_id" not in session:
112 |         return redirect("/login")
113 |     if request.method == "POST":
114 |         img = request.files["image"]
115 |         img.save(os.path.join(UPLOADS, img.filename))
116 |         c = db(); cur = c.cursor()
117 |         cur.execute("""
118 |             INSERT INTO harvest(farmer_id,quantity,expected_price,image,harvest_date)
119 |             VALUES (?,?,?,?,?,?)
120 |             """, (
121 |                 session["farmer_id"],
122 |                 request.form["quantity"],
123 |                 request.form["price"],
124 |                 img.filename,
125 |                 request.form["date"]
126 |             ))
127 |         c.commit(); c.close()
128 |         return redirect("/dashboard")
129 |     return render_template("upload_harvest.html")
130 |
131 | @app.route("/my harvests")
132 | def my_harvests():
133 |     if "farmer_id" not in session:
134 |         return redirect("/login")
135 |     c = db(); cur = c.cursor()
136 |     cur.execute("SELECT * FROM harvest WHERE farmer_id=?", (session["farmer_id"],))
137 |     rows = cur.fetchall()
138 |     c.close()
139 |     return render_template("my_harvests.html", harvests=rows)
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
140 |
141 | @app.route("/edit_harvest/<int:id>", methods=[ "GET", "POST" ])
142 | def edit_harvest(id):
143 |     if "farmer_id" not in session:
144 |         return redirect("/login")
145 |
146 |     c = db(); cur = c.cursor()
147 |     cur.execute(
148 |         "SELECT * FROM harvest WHERE id=? AND farmer_id=?",
149 |         (id, session["farmer_id"]))
150 |     )
151 |     h = cur.fetchone()
152 |
153 |     if not h:
154 |         c.close()
155 |         return "Unauthorized", 403
156 |
157 |     if request.method == "POST":
158 |         cur.execute("""
159 |             UPDATE harvest
160 |                 SET quantity=?, expected_price=?, harvest_date=?, status='pending'
161 |                 WHERE id=?
162 |             """ , (
163 |                 request.form["quantity"],
164 |                 request.form["price"],
165 |                 request.form["date"],
166 |                 id
167 |             ))
168 |         c.commit(); c.close()
169 |         return redirect("/my_harvests")
170 |
171 |     c.close()
172 |     return render_template("edit_harvest.html", harvest=h)
173 |
174 | @app.route("/inbox")
175 | def inbox():
176 |     if "farmer_id" not in session:
177 |         return redirect("/login")
178 |     c = db(); cur = c.cursor()
179 |     cur.execute("""
180 |         SELECT d.*, h.quantity, h.expected_price
181 |         FROM deal d JOIN harvest h ON d.harvest_id=h.id
182 |         WHERE h.farmer_id=?
183 |         ORDER BY d.created_at DESC
184 |         """ , (session["farmer_id"],))
185 |     deals = cur.fetchall()
186 |     c.close()
187 |     return render_template("inbox.html", deals=deals)
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
188 |
189 | # ===== BUYER =====
190 | @app.route("/buyer")
191 | def buyer():
192 |     min_p = request.args.get("min_price")
193 |     max_p = request.args.get("max_price")
194 |     min_q = request.args.get("min_qty")
195 |
196 |     q = """
197 |     SELECT h.*, f.name, f.trust_tier
198 |     FROM harvest h JOIN farmer f ON h.farmer_id=f.id
199 |     WHERE h.status='approved'
200 |     """
201 |     params = []
202 |
203 |     if min_p:
204 |         q += " AND CAST(h.expected_price AS INTEGER)>=?"
205 |         params.append(min_p)
206 |     if max_p:
207 |         q += " AND CAST(h.expected_price AS INTEGER)<=?"
208 |         params.append(max_p)
209 |     if min_q:
210 |         q += " AND CAST(h.quantity AS INTEGER)>=?"
211 |         params.append(min_q)
212 |
213 |     c = db(); cur = c.cursor()
214 |     cur.execute(q, params)
215 |     rows = cur.fetchall()
216 |     c.close()
217 |     return render_template("buyer_dashboard.html", harvests=rows)
218 |
219 | # ===== DEAL =====
220 | @app.route("/initiate deal/<int:hid>", methods=["POST"])
221 | def initiate_deal(hid):
222 |     c = db(); cur = c.cursor()
223 |     cur.execute("SELECT quantity, expected_price FROM harvest WHERE id=?", (hid,))
224 |     h = cur.fetchone()
225 |     value = float(h["quantity"]) * 1000 * float(h["expected_price"])
226 |     commission = value * 0.02
227 |     net = value - commission
228 |     cur.execute("""
229 |         INSERT INTO deal(harvest_id,buyer_name,buyer_email,
230 |             deal_value,commission,farmer_net,status,created_at)
231 |             VALUES (?,?,?,?,?,?,'initiated',?)
232 |         """, (
233 |             hid,
234 |             request.form["name"],
235 |             request.form["email"],
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
236 |         value, commission, net,
237 |         datetime.now().strftime("%Y %m %d %H:%M")
238 |     ))
239 |     c.commit(); c.close()
240 |     return redirect("/buyer")
241 |
242 | @app.route("/deal/delivered/<int:id>")
243 | def delivered(id):
244 |     if "farmer_id" not in session:
245 |         return redirect("/login")
246 |     c = db(); cur = c.cursor()
247 |     cur.execute("UPDATE deal SET status='delivered' WHERE id=?", (id,))
248 |     c.commit(); c.close()
249 |     return redirect("/inbox")
250 |
251 | @app.route("/deal/complete/<int:id>")
252 | def complete(id):
253 |     if not session.get("admin"):
254 |         return redirect("/admin/login")
255 |     c = db(); cur = c.cursor()
256 |     cur.execute("UPDATE deal SET status='completed' WHERE id=?", (id,))
257 |     c.commit(); c.close()
258 |     return redirect("/admin/dashboard")
259 |
260 | # ===== ADMIN =====
261 | @app.route("/admin/login", methods=["GET", "POST"])
262 | def admin_login():
263 |     if request.method=="POST":
264 |         if request.form["email"]=="admin@kespo.com" and request.form["password"]=="admin123":
265 |             session.clear()
266 |             session["admin"]=True
267 |             return redirect("/admin/dashboard")
268 |     return render_template("login.html")
269 |
270 | @app.route("/admin/dashboard")
271 | def admin_dashboard():
272 |     if not session.get("admin"):
273 |         return redirect("/admin/login")
274 |     c = db(); cur = c.cursor()
275 |     cur.execute("""
276 |         SELECT h.*, f.name FROM harvest h
277 |         JOIN farmer f ON h.farmer_id=f.id
278 |         WHERE h.status='pending'
279 |         """
280 |     )
281 |     harvests = cur.fetchall()
282 |     cur.execute("SELECT * FROM deal WHERE status='delivered'")
283 |     deals = cur.fetchall()
284 |     c.close()
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
284 |     return render_template("admin_dashboard.html", harvests=harvests, deals=deals)
285 |
286 | @app.route("/admin/approve/<int:id>")
287 | def approve(id):
288 |     if not session.get("admin"):
289 |         return redirect("/admin/login")
290 |     c = db(); cur = c.cursor()
291 |     cur.execute("UPDATE harvest SET status='approved' WHERE id=?", (id,))
292 |     c.commit(); c.close()
293 |     return redirect("/admin/dashboard")
294 |
295 | # ===== RUN =====
296 | if __name__=="__main__":
297 |     init_db()
298 |     app.run(port=5001, debug=True)
299 |
```

File: generate_docs.py

```
1 |
```

File: generate_docs.py

```
1 | import os
2 | from fpdf import FPDF
3 | from datetime import datetime
4 | from pathlib import Path
5 |
6 | class SimplePDF(FPDF):
7 |     def header(self):
8 |         self.set_font('Arial', 'B', 12)
9 |         self.cell(0, 10, 'KESPO Project Code Documentation', 0, 1, 'C')
10 |        self.cell(0, 10, f'Generated on {datetime.now().strftime("%Y %m %d %H:%M:%S")}', 0, 1, 'C')
11 |        self.ln(10)
12 |
13 |     def chapter_title(self, title):
14 |         self.set_font('Arial', 'B', 12)
15 |         self.set_fill_color(200, 220, 255)
16 |         self.cell(0, 10, title, 0, 1, 'L', 1)
17 |         self.ln(4)
18 |
19 |     def sanitize_text(text):
20 |         """Replace or remove problematic Unicode characters"""
21 |         if not text:
22 |             return ""
23 |
24 |         # Common Unicode replacements
25 |         replacements = {
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
26 |         'Rs.': 'Rs.',      # Indian Rupee symbol
27 |         '‐': '‐',        # en dash
28 |         '‐‐': '‐‐',      # em dash
29 |         '“’: ‘“’,       # left double quote
30 |         '”’: ‘”’,       # right double quote
31 |         '‘’: ‘‘’,       # left single quote
32 |         '‘’: ‘‘’,       # right single quote
33 |         '…’: ‘…’,       # ellipsis
34 |         '‐‐‐’: ‘‐‐‐’,    # horizontal bar
35 |         '‐‐‐‐’: ‘‐‐‐‐’,  # narrow no break space
36 |         '< ': '< ',       # leftwards arrow
37 |         '> ': '> ',       # rightwards arrow
38 |         '(up)': '(up)',   # upwards arrow
39 |         '(down)': '(down)', # downwards arrow
40 |     }
41 |
42 |     # Apply replacements
43 |     for k, v in replacements.items():
44 |         text = text.replace(k, v)
45 |
46 |     # Remove any remaining non ASCII characters
47 |     return ''.join(c if ord(c) < 128 and c not in '\x00 \x1f\x7f \x9f' else ' '
48 |                         for c in text)
49 |
50 | def get_file_content(filepath):
51 |     try:
52 |         with open(filepath, 'r', encoding='utf 8') as file:
53 |             return sanitize_text(file.read())
54 |     except Exception as e:
55 |         try:
56 |             with open(filepath, 'r', encoding='latin 1') as file:
57 |                 return sanitize_text(file.read())
58 |         except Exception as e:
59 |             print(f"Error reading {filepath}: {str(e)}")
60 |             return f"[Error reading file: {str(e)}]"
61 |
62 | def create_pdf_documentation(project_path, output_file):
63 |     pdf = SimplePDF()
64 |     pdf.set_auto_page_break(auto=True, margin=15)
65 |     pdf.set_font('Courier', '', 8)  # Use built in Courier font
66 |
67 |     added_sections = set()
68 |
69 |     file_types = {
70 |         'Python': ['.py'],
71 |         'HTML': ['.html'],
72 |         'CSS': ['.css'],
73 |         'JavaScript': ['.js']
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
74 |     }
75 |
76 |     for root, _, files in os.walk(project_path):
77 |         if any(x in root for x in ['venv', '__pycache__', '.git', '.idea', 'node_modules']):
78 |             continue
79 |
80 |         for file in sorted(files):
81 |             file_path = Path(root) / file
82 |             rel_path = file_path.relative_to(project_path)
83 |             file_ext = file_path.suffix.lower()
84 |
85 |             file_type = None
86 |             for ftype, exts in file_types.items():
87 |                 if file_ext in exts:
88 |                     file_type = ftype
89 |                     break
90 |
91 |             if file_type:
92 |                 try:
93 |                     if file_type not in added_sections:
94 |                         pdf.add_page()
95 |                         pdf.chapter_title(f"{file_type} Files")
96 |                         added_sections.add(file_type)
97 |
98 |                     content = get_file_content(file_path)
99 |                     pdf.set_font('Arial', 'B', 10)
100 |                     pdf.cell(0, 10, f"File: {rel_path}", 0, 1)
101 |                     pdf.set_font('Courier', '', 8)
102 |
103 |                     lines = content.split('\n')
104 |                     for i, line in enumerate(lines, 1):
105 |                         try:
106 |                             pdf.cell(0, 5, f"{i:4} | {line}", 0, 1)
107 |                         except:
108 |                             continue
109 |
110 |                         pdf.ln(5)
111 |                     except Exception as e:
112 |                         print(f"Error processing {file_path}: {str(e)}")
113 |                         continue
114 |
115 |                 try:
116 |                     pdf.output(output_file)
117 |                     print(f"Successfully created: {output_file}")
118 |                     return True
119 |                 except Exception as e:
120 |                     print(f"Error saving PDF: {str(e)}")
121 |                     alt_output = os.path.join(os.path.expanduser('~'), 'Desktop', 'kespo_documentation.pdf')
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
122 |         try:
123 |             pdf.output(alt_output)
124 |             print(f"PDF saved to alternative location: {alt_output}")
125 |             return True
126 |         except Exception as e2:
127 |             print(f"Failed to save PDF: {str(e2)}")
128 |             return False
129 |
130 | if __name__ == "__main__":
131 |     project_path = os.path.dirname(os.path.abspath(__file__))
132 |     output_file = os.path.join(project_path, "kespo_documentation.pdf")
133 |
134 |     print(f"Generating documentation PDF to: {output_file}")
135 |     if create_pdf_documentation(project_path, output_file):
136 |         print("Documentation generated successfully!")
137 |     else:
138 |         print("Failed to generate documentation. Please check the error messages above.")
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

CSS Files

File: static/style.css

```
1 | /* ====== RESET ====== */
2 | *
3 |   box-sizing: border-box;
4 |   font-family: apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, Arial, sans-serif;
5 |
6 |
7 | body {
8 |   margin: 0;
9 |   padding: 0;
10 |  background: #f4f6f9;
11 |  color: #333;
12 | }
13 |
14 | /* ====== GLOBAL ====== */
15 | h2, h3 {
16 |   margin: 16px 0;
17 |   color: #2c3e50;
18 | }
19 |
20 | a {
21 |   text-decoration: none;
22 |   color: #2e7d32;
23 |   font-weight: 500;
24 | }
25 |
26 | a:hover {
27 |   text-decoration: underline;
28 | }
29 |
30 | /* ====== LAYOUT ====== */
31 | .container {
32 |   max-width: 1100px;
33 |   margin: auto;
34 |   padding: 20px;
35 | }
36 |
37 | /* ====== CARD ====== */
38 | .card {
39 |   background: #fff;
40 |   border-radius: 12px;
41 |   padding: 16px;
42 |   margin-bottom: 18px;
43 |   box-shadow: 0 8px 20px rgba(0,0,0,0.05);
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
44 | }
45 |
46 | /* ===== FORMS ===== */
47 | form input,
48 | form textarea,
49 | form button {
50 |   width: 100%;
51 |   padding: 10px 12px;
52 |   margin top: 6px;
53 |   margin bottom: 12px;
54 |   border radius: 8px;
55 |   border: 1px solid #ccc;
56 |   font size: 14px;
57 | }
58 |
59 | form input:focus,
60 | form textarea:focus {
61 |   outline: none;
62 |   border color: #2e7d32;
63 | }
64 |
65 | form button {
66 |   background: #2e7d32;
67 |   color: white;
68 |   border: none;
69 |   cursor: pointer;
70 |   font size: 15px;
71 | }
72 |
73 | form button:hover {
74 |   background: #256428;
75 | }
76 |
77 | /* ===== NAV ===== */
78 | .nav {
79 |   margin top: 20px;
80 | }
81 |
82 | .nav a {
83 |   display: inline block;
84 |   margin: 8px 0;
85 | }
86 |
87 | /* ===== IMAGES ===== */
88 | img {
89 |   border radius: 10px;
90 |   max width: 100%;
91 | }
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
92 |
93 | /* ===== FILTER BAR ===== */
94 | .filter bar {
95 |   display: flex;
96 |   gap: 10px;
97 |   flex wrap: wrap;
98 | }
99 |
100| .filter bar input {
101|   flex: 1;
102| }
103|
104| /* ===== STATUS TAGS ===== */
105| .status {
106|   display: inline block;
107|   padding: 4px 10px;
108|   border radius: 20px;
109|   font size: 12px;
110|   font weight: bold;
111| }
112|
113| .status.pending {
114|   background: #fff3cd;
115|   color: #856404;
116| }
117|
118| .status.approved {
119|   background: #d4edda;
120|   color: #155724;
121| }
122|
123| .status.delivered {
124|   background: #cce5ff;
125|   color: #004085;
126| }
127|
128| .status.completed {
129|   background: #e2e3e5;
130|   color: #383d41;
131| }
132|
133| /* ===== ADMIN ===== */
134| .admin actions a {
135|   display: inline block;
136|   margin top: 10px;
137|   padding: 8px 12px;
138|   background: #1976d2;
139|   color: white;
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
140 |   border radius: 6px;
141 | }
142 |
143 | .admin actions a:hover {
144 |   background: #155fa0;
145 | }
146 |
147 | /* ===== FOOT LINKS ===== */
148 | .footer link {
149 |   margin top: 20px;
150 |   display: inline block;
151 | }
152 |
153 | /* ===== MOBILE ===== */
154 | @media (max width: 768px) {
155 |   .container {
156 |     padding: 12px;
157 |   }
158 | }
159 |
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

HTML Files

File: templates/admin_dashboard.html

```
1 | <h2>Admin Panel</h2>
2 |
3 | <h3>Pending Harvests</h3>
4 | {% for h in harvests %}
5 | <div>
6 | {{ h.name }} {{ h.quantity }} Tons
7 | <a href="/admin/approve/{{ h.id }}">Approve</a>
8 | </div>
9 | {% endfor %}
10 |
11 | <h3>Delivered Deals</h3>
12 | {% for d in deals %}
13 | <div>
14 | {{ d.buyer_name }} Rs.{{ d.deal_value }}
15 | <a href="/deal/complete/{{ d.id }}">Release</a>
16 | </div>
17 | {% endfor %}
18 |
```

File: templates/admin_login.html

```
1 | <!DOCTYPE html>
2 | <html>
3 | <head>
4 |     <title>KESPO | Admin Login</title>
5 |     <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}>
6 | </head>
7 | <body>
8 |     <h2>KESPO Admin Login</h2>
9 |
10 |     {% with messages = get_flashed_messages(with_categories=true) %}
11 |         {% if messages %}
12 |             {% for category, message in messages %}
13 |                 <div class="alert alert {{ category }}>{{ message }}</div>
14 |             {% endfor %}
15 |         {% endif %}
16 |     {% endwith %}
17 |
18 |     <form method="post">
19 |         <div>
20 |             <label for="email">Email:</label>
21 |             <input type="email" id="email" name="email" required>
22 |         </div>
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
23 |         <div>
24 |             <label for="password">Password:</label>
25 |             <input type="password" id="password" name="password" required>
26 |         </div>
27 |         <button type="submit">Login</button>
28 |     </form>
29 | </body>
30 | </html>
31 | 
```

File: templates/buyer_dashboard.html

```
1 | <h2>Buyer Marketplace</h2>
2 |
3 | <form method="get">
4 | Min Price <input name="min_price">
5 | Max Price <input name="max_price">
6 | Min Qty <input name="min_qty">
7 | <button type="submit">Filter</button>
8 | </form>
9 |
10 | {% for h in harvests %}
11 | <div>
12 |     <br>
13 |     {{ h.name }} ({{ h.trust_tier }})<br>
14 |     {{ h.quantity }} Tons @ Rs.{{ h.expected_price }}/KG
15 |     <form method="post" action="/initiate deal/{{ h.id }}">
16 |         <input name="name" placeholder="Your name" required>
17 |         <input name="email" placeholder="Your email" required>
18 |         <button type="submit">Deal</button>
19 |     </form>
20 | </div>
21 | {% endfor %}
22 |
23 | <a href="/dashboard">Back</a>
24 | 
```

File: templates/dashboard.html

```
1 | <h2>Welcome {{ farmer_name }}</h2>
2 | <p>Trust Tier: {{ trust_tier }}</p>
3 | <p>Total Harvests: {{ harvest_count }}</p>
4 | <a href="/upload">Upload Harvest</a><br>
5 | <a href="/my harvests">My Harvests</a><br>
6 | <a href="/inbox">Inbox</a><br>
7 | <a href="/buyer">Buyer Marketplace</a><br>
8 | <a href="/logout">Logout</a>
9 | 
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

File: templates/edit_harvest.html

```
1 | <!DOCTYPE html>
2 | <html>
3 | <head>
4 |   <title>Edit Harvest</title>
5 | </head>
6 | <body>
7 |
8 |   <h2>Edit Harvest Details</h2>
9 |
10 | <form method="post">
11 |   <label>Quantity (Tons)</label><br>
12 |   <input type="number" step="0.1" name="quantity"
13 |     value="{{ harvest.quantity }}" required><br><br>
14 |
15 |   <label>Expected Price (Rs. / KG)</label><br>
16 |   <input type="number" name="price"
17 |     value="{{ harvest.expected_price }}" required><br><br>
18 |
19 |   <label>Harvest Date</label><br>
20 |   <input type="date" name="date"
21 |     value="{{ harvest.harvest_date }}" required><br><br>
22 |
23 |   <button type="submit">Save Changes</button>
24 | </form>
25 |
26 | <br>
27 | <p><b>Note:</b> Any edit will send this harvest back for admin approval.</p>
28 |
29 | <a href="/my harvests">Cancel</a>
30 |
31 | </body>
32 | </html>
33 |
```

File: templates/inbox.html

```
1 | <!DOCTYPE html>
2 | <html>
3 | <head>
4 |   <title>KESPO | Farmer Inbox</title>
5 |   <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
6 | </head>
7 | <body>
8 |
9 |   <h2>Deal Inbox</h2>
10 |
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
11 | {%- if deals|length == 0 %}  
12 |     <p>No deals yet.</p>  
13 | {%- endif %}  
14 |  
15 | {%- for d in deals %}  
16 | <div class="card">  
17 |  
18 |     <b>Buyer Name:</b> {{ d.buyer_name }}<br>  
19 |     <b>Buyer Email:</b> {{ d.buyer_email }}<br><br>  
20 |  
21 |     <b>Quantity:</b> {{ d.quantity }} Tons<br>  
22 |     <b>Price:</b> Rs. {{ d.expected_price }} / KG<br><br>  
23 |  
24 |     <b>Deal Value:</b> Rs. {{ d.deal_value }}<br>  
25 |     <b>KESPO Fee (2%):</b> Rs. {{ d.commission }}<br>  
26 |     <b>You Receive:</b> Rs. {{ d.farmer_net }}<br><br>  
27 |  
28 |     <b>Status:</b>  
29 |     <span class="badge">{{ d.status | upper }}</span><br><br>  
30 |  
31 |     <! FARMER ACTION >  
32 |     {% if d.status != 'delivered' and d.status != 'completed' %}  
33 |         <a href="/deal/delivered/{{ d.id }}" class="btn">  
34 |             Mark Delivered  
35 |         </a>  
36 |     {% endif %}  
37 |  
38 | </div>  
39 | {%- endfor %}  
40 |  
41 | <br>  
42 | <a href="/dashboard"> Back to Dashboard</a>  
43 |  
44 | </body>  
45 | </html>  
46 |
```

File: templates/login.html

```
1 | <!DOCTYPE html>  
2 | <html>  
3 | <head>  
4 |     <title>KESPO | Farmer Login</title>  
5 |     <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}>  
6 | </head>  
7 | <body>  
8 |  
9 | <h2>KESPO Farmer Login</h2>
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
10 |
11 | <form method="post">
12 |     Email:
13 |     <input type="email" name="email" required>
14 |
15 |     Password:
16 |     <input type="password" name="password" required>
17 |
18 |     <button type="submit">Login</button>
19 | </form>
20 |
21 | <br>
22 | <a href="/register">New farmer? Register</a>
23 |
24 | </body>
25 | </html>
26 |
```

File: templates/my_harvests.html

```
1 | <!DOCTYPE html>
2 | <html>
3 | <head>
4 |     <title>KESPO | My Harvests</title>
5 |     <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
6 | </head>
7 | <body>
8 |
9 | <h2>My Harvest Listings</h2>
10 |
11 | <a href="/dashboard">< Back to Dashboard</a><br><br>
12 |
13 | {% if harvests|length == 0 %}
14 | <p>You have not uploaded any harvests yet.</p>
15 | {% endif %}
16 |
17 | {% for h in harvests %}
18 | <div class="card">
19 |     <br><br>
20 |
21 |     <b>Quantity:</b> {{ h.quantity }} Tons<br>
22 |     <b>Price:</b> Rs. {{ h.expected_price }} per KG<br>
23 |     <b>Deal Value:</b> Rs. {{ h.deal_value }}<br>
24 |     <b>Harvest Date:</b> {{ h.harvest_date }}<br>
25 |     <b>Status:</b> {{ h.status | upper }}<br>
26 |
27 |     {% if h.rejection_reason %}
28 |         <p style="color:red;">
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
29 |             <b>Rejection Reason:</b> {{ h.rejection_reason }}
```

```
30 |         </p>
```

```
31 |     {%- endif %}
```

```
32 | 
```

```
33 |     <br>
```

```
34 |     <a href="/edit harvest/{{ h.id }}"> Edit</a> |
```

```
35 |     <a href="/delete harvest/{{ h.id }}" onclick="return confirm('Delete this harvest permanently?')">
```

```
36 |         Delete
```

```
37 |     </a>
```

```
38 | </div>
```

```
39 | {%- endfor %}
```

```
40 | 
```

```
41 | </body>
```

```
42 | </html>
```

```
43 | 
```

File: templates/register.html

```
1 | <!DOCTYPE html>
```

```
2 | <html>
```

```
3 | <head>
```

```
4 |     <title>KESPO | Farmer Registration</title>
```

```
5 |     <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}>
```

```
6 | </head>
```

```
7 | <body>
```

```
8 | 
```

```
9 | <h2>KESPO    Farmer Registration</h2>
```

```
10 | 
```

```
11 | <form method="post">
```

```
12 |     Name:
```

```
13 |     <input type="text" name="name" required>
```

```
14 | 
```

```
15 |     Email:
```

```
16 |     <input type="email" name="email" required>
```

```
17 | 
```

```
18 |     Phone:
```

```
19 |     <input type="text" name="phone" required>
```

```
20 | 
```

```
21 |     Password:
```

```
22 |     <input type="password" name="password" required>
```

```
23 | 
```

```
24 |     <button type="submit">Register</button>
```

```
25 | </form>
```

```
26 | 
```

```
27 | <br>
```

```
28 | <a href="/login">Already registered? Login</a>
```

```
29 | 
```

```
30 | </body>
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
31 | </html>
32 |
```

File: templates/send_message.html

```
1 | <!DOCTYPE html>
2 | <html>
3 | <head>
4 |   <title>Contact Farmer</title>
5 |   <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
6 | </head>
7 | <body>
8 |
9 | <h2>Send Inquiry</h2>
10 |
11 | <form method="post">
12 |   Your Name:
13 |   <input name="name" required>
14 |
15 |   Email:
16 |   <input type="email" name="email" required>
17 |
18 |   Message:
19 |   <textarea name="message" required></textarea>
20 |
21 |   <button type="submit">Send</button>
22 | </form>
23 |
24 | <a href="/buyer">Back</a>
25 |
26 | </body>
27 | </html>
28 |
```

File: templates/upload_harvest.html

```
1 | <!DOCTYPE html>
2 | <html>
3 | <head>
4 |   <title>KESPO | Upload Harvest</title>
5 |   <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
6 | </head>
7 | <body>
8 |
9 | <h2>KESPO Upload Pomegranate Harvest</h2>
10 |
11 | <form method="post" enctype="multipart/form data">
12 |
```

KESPO Project Code Documentation

Generated on 2025-12-22 12:39:53

```
13 |     Quantity (in Tons):  
14 |     <input type="number" step="0.1" name="quantity" required>  
15 |  
16 |     Expected Price (Rs. per KG):  
17 |     <input type="number" name="price" required>  
18 |  
19 |     Harvest Date:  
20 |     <input type="date" name="date" required>  
21 |  
22 |     Harvest Image:  
23 |     <input type="file" name="image" required>  
24 |  
25 |     <button type="submit">Upload Harvest</button>  
26 | </form>  
27 |  
28 | <br>  
29 | <a href="/dashboard">Back to Dashboard</a>  
30 |  
31 | </body>  
32 | </html>  
33 |
```