

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Card Swiper</title>
  <link rel="stylesheet" type="text/css" href="css/jTinder.css">
  <style>
    .floating-button {
      position: fixed;
      bottom: 20px;
      padding: 12px 24px;
      border-radius: 25px;
      text-decoration: none;
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
      transition: all 0.3s ease;
      display: flex;
      align-items: center;
      font-family: Arial, sans-serif;
      font-size: 16px;
      color: white;
    }

    .floating-button:hover {
      box-shadow: 0 6px 12px rgba(0, 0, 0, 0.15);
      transform: translateY(-2px);
      color: white;
      text-decoration: none;
    }

    .floating-button i {
      margin-right: 8px;
      font-size: 20px;
    }

    .liked-button {
      right: 20px;
      background-color: #007bff;
    }
  </style>
</head>
</html>
```

```

.liked-button:hover {
    background-color: #0056b3;
}

.avatar-button {
    left: 20px;
    background-color: #28a745;
}

.avatar-button:hover {
    background-color: #218838;
}

.wrap {
    display: flex;
    justify-content: center;
    align-items: center;
    height: 360px;
    width: 300px;
    margin: 0 auto;
    padding-left: 5px;
}
</style>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
</head>
<body>

<div class="wrap" style="height: 360px; width: 233px; ">
    <div id="tinderslide">
        <ul id="cardContainer">
            <!-- Cards will be dynamically added here -->
        </ul>
    </div>
</div>

<div class="actions">

```

```
<a href="#" class="dislike"><i></i></a>
<a href="#" class="save"><i></i></a>
<a href="#" class="like"><i></i></a>
</div>
```

```
<a href="liked-images.html" class="floating-button liked-button">
  <i class="fas fa-heart"></i>
  Liked Cards
</a>
```

```
<a href="select-avatar.html" class="floating-button avatar-button">
  <i class="fas fa-user-circle"></i>
  Archetypes
</a>
```

```
<div id="status"></div>
```

```
<!-- Scripts -->
```

```
<script type="text/javascript" src="js/jquery.min.js"></script>
<script type="text/javascript" src="js/jquery.transform2d.js"></script>
<script type="text/javascript" src="js/jquery.jTinder.js"></script>
```

```
<script>
```

```
document.addEventListener('DOMContentLoaded', () => {
  const username = localStorage.getItem('username');
  if (!username) {
    alert('No user logged in! Redirecting...');
    window.location.href = 'login.html';
    return;
  }
}
```

```
// Fetch cards from backend
```

```
fetch('/app-data')
  .then(response => response.json())
  .then(data => {
    const cardContainer = document.getElementById('cardContainer');
```

```
// Find current user and their selected skills
```

```
const currentUser = data.users.find(user => user.name === username);
const userSelectedSkills = currentUser.selectedSkills || [];
```

```

    if (!userSelectedSkills.length) {
      cardContainer.innerHTML = '<p>No skills selected. Please select skills
first.</p>';
      return;
    }

    // Filter cards based on user's selected skills
    const matchingCards = data.allCards.filter(card =>
      // Check if any of the card's skills match user's selected skills
      card.skill.some(cardSkill =>
        userSelectedSkills.includes(cardSkill)
      )
    );

    // Clear existing cards
    cardContainer.innerHTML = "";

    if (matchingCards.length === 0) {
      cardContainer.innerHTML = '<p>No matching cards found for your
skills.</p>';
      return;
    }

    // Add filtered cards
    matchingCards.forEach((card, index) => {
      const li = document.createElement('li');
      li.className = `pane5`;
      li.innerHTML = `
        <div class="img" style="background-image:
url('img/pane/${card.backgroundImage}')"></div>

        <div class="like"></div>
        <div class="dislike"></div>
      `;
      cardContainer.appendChild(li);
    });

    // Initialize jTinder only if we have cards
    if (matchingCards.length > 0) {

```

```

$("#tinderslide").jTinder({
  onDislike: function (item) {
    // Handle dislike
  },
  onLike: function (item) {
    // Get the liked card data
    const index = item.index();
    const likedCard = matchingCards[index];

    // Save liked card
    saveLikedCard(likedCard);
  }
});

// Bind buttons
$('.actions .like, .actions .dislike').click(function(e){
  e.preventDefault();
  $("#tinderslide").jTinder($(this).hasClass('like') ? 'like' : 'dislike');
});
}
});

// Function to save liked card
function saveLikedCard(card) {
  fetch('/app-data')
    .then(response => response.json())
    .then(data => {
      // Find current user
      const users = data.users || [];
      const currentUserIndex = users.findIndex(user => user.name ===
username);

      if (currentUserIndex !== -1) {
        // Initialize likedCards array if it doesn't exist
        if (!users[currentUserIndex].likedCards) {
          users[currentUserIndex].likedCards = [];
        }

        // Add card title to user's likedCards if not already present
        if (!users[currentUserIndex].likedCards.includes(card.title)) {

```

```
        users[currentUserIndex].likedCards.push(card.title);
    }

    // Prepare updated data
    const updatedData = {
        ...data,
        users: users
    };

    // Save back to backend
    return fetch('/app-data', {
        method: 'POST',
        headers: {
            'Content-Type': 'application/json'
        },
        body: JSON.stringify(updatedData)
    });
}
})
.catch(error => console.error('Error saving liked card:', error));
}
});
</script>
</body>
</html>
```