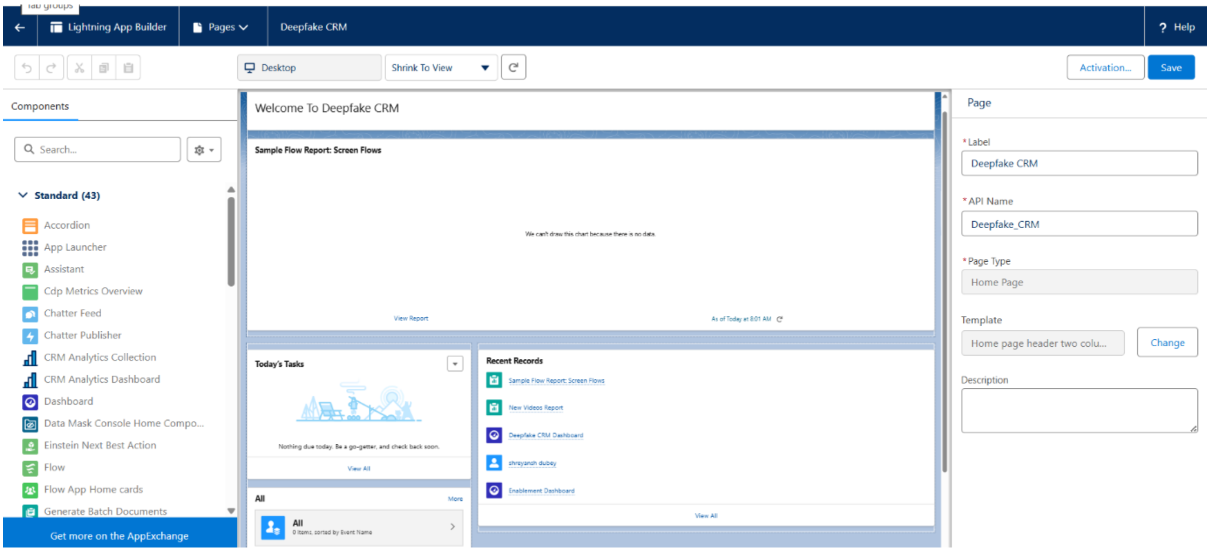
Phase 6: User Interface Development – Salesforce

1. Lightning App Builder

- Drag-and-drop tool to create **App, Home, or Record pages**.
- Types of pages:
  - **App Page:** Standalone page in App Launcher.
  - **Home Page:** Landing page with dashboards, reports, and components.
  - **Record Page:** Custom layout for specific objects (e.g., Video).
- Features:
  - Dynamic visibility
  - Tabs for organization
  - Custom LWC integration
- **Tip:** Use tabs template for Record Pages to organize Details, Related Lists, and custom components.



Welcome To Deepfake CRM

Sample Flow Report: Screen Flows

We can't draw this chart because there is no data.

As of Today at 8:01 AM

Today's Tasks

Nothing due today. Be a go-getter, and check back soon.

All

All
0 items, sorted by Event Name

Recent Records

- Sample Flow Report: Screen Flows
- New Videos Report
- Deepfake CRM Dashboard
- shreyansh dubey
- Enablement Dashboard

Page

* Label: Deepfake CRM

* API Name: Deepfake_CRM

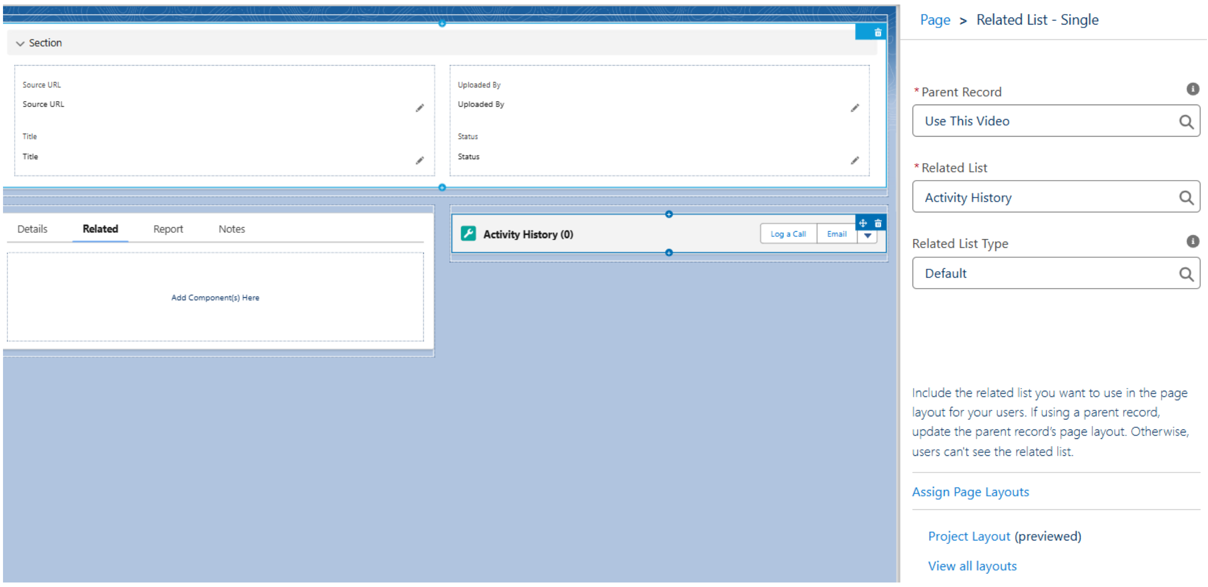
* Page Type: Home Page

Template: Home page header two colu...

Description:

2. Record Pages

- Customizes layout for **object records**.
- Include:
  - **Highlights Panel:** Key fields at top.
  - **Tabs:** Organize Details, Related Lists, and Custom LWCs.
  - **Custom Components:** Video Player, Alerts, Detection status.
- Assign pages to **profiles or apps**.



* Parent Record: Use This Video

* Related List: Activity History

Related List Type: Default

Include the related list you want to use in the page layout for your users. If using a parent record, update the parent record's page layout. Otherwise, users can't see the related list.

Assign Page Layouts

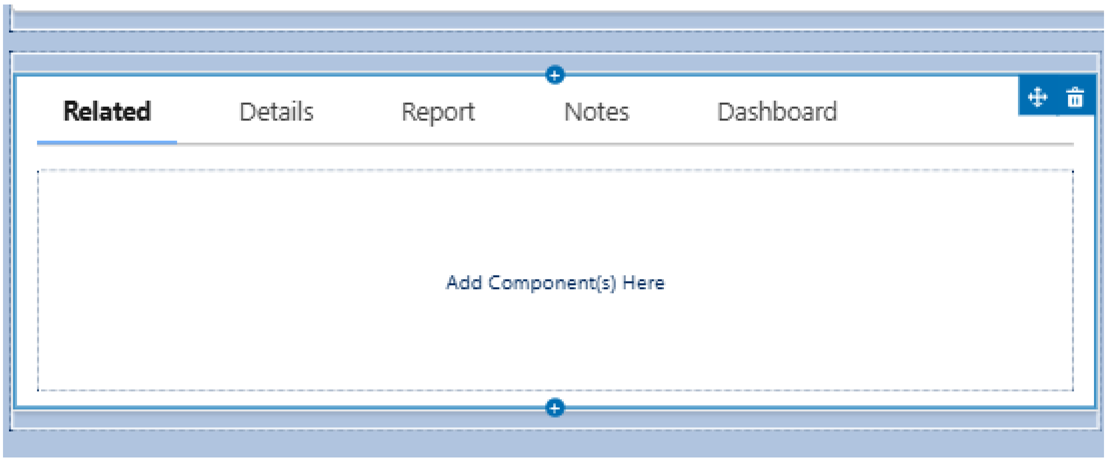
Project Layout (previewed)

View all layouts

---

3. Tabs

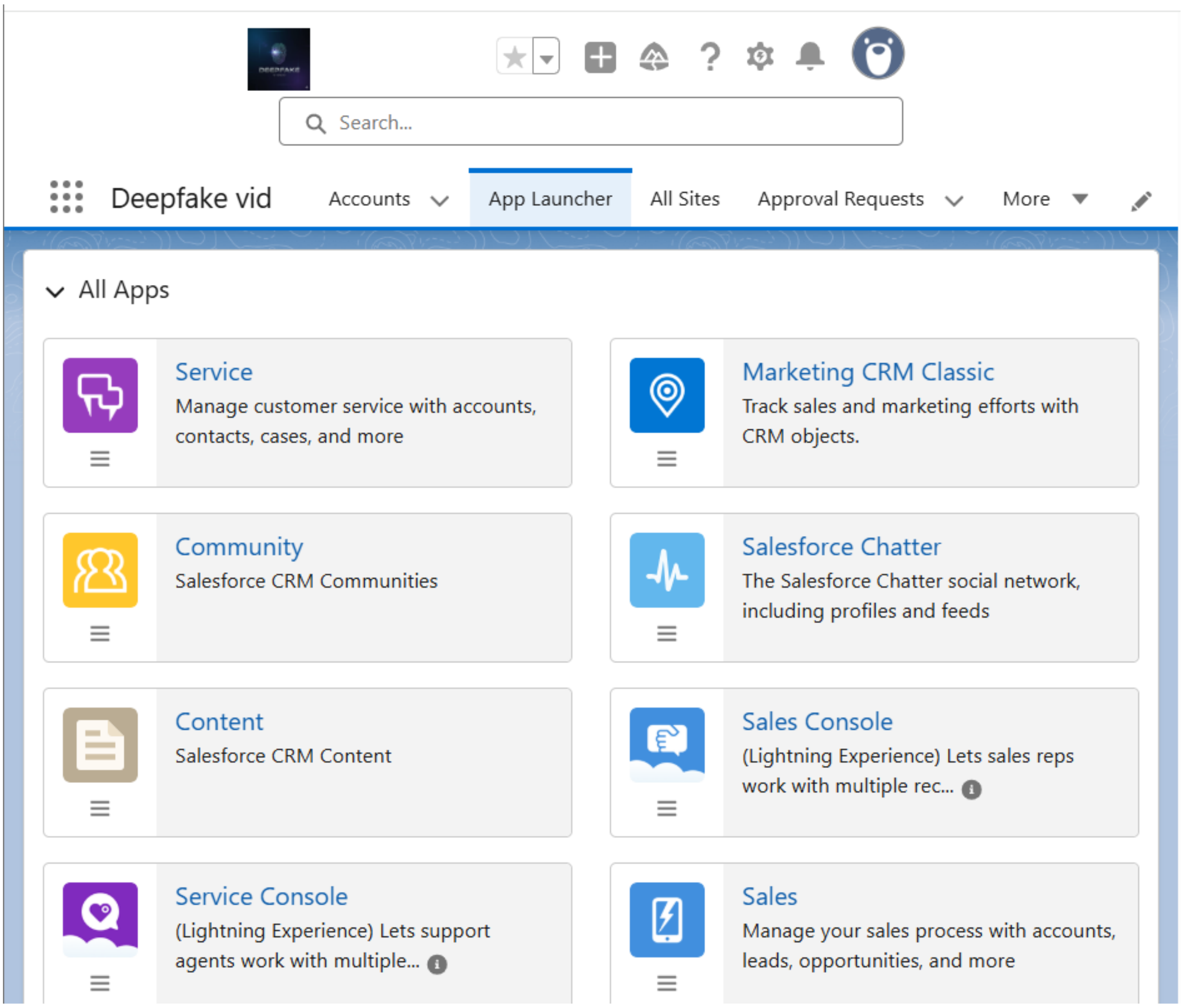
- Organize information efficiently.
- Examples for Video object:
  - Details (Record Fields)
  - Related (Alerts, Comments)
  - Video Player (LWC)
  - Detection/Info (Custom LWC or Rich Text)
  - Report
  - Notes
  - Dashboard



---

4. Home Page Layouts

- Custom landing page for users.
- Components:
  - Reports, Dashboards
  - Announcements
  - Custom LWC (e.g., Recent Videos)
- Can assign different layouts for different **profiles**.



All Apps

Service
Manage customer service with accounts, contacts, cases, and more

Marketing CRM Classic
Track sales and marketing efforts with CRM objects.

Community
Salesforce CRM Communities

Salesforce Chatter
The Salesforce Chatter social network, including profiles and feeds

Content
Salesforce CRM Content

Sales Console
(Lightning Experience) Lets sales reps work with multiple rec...

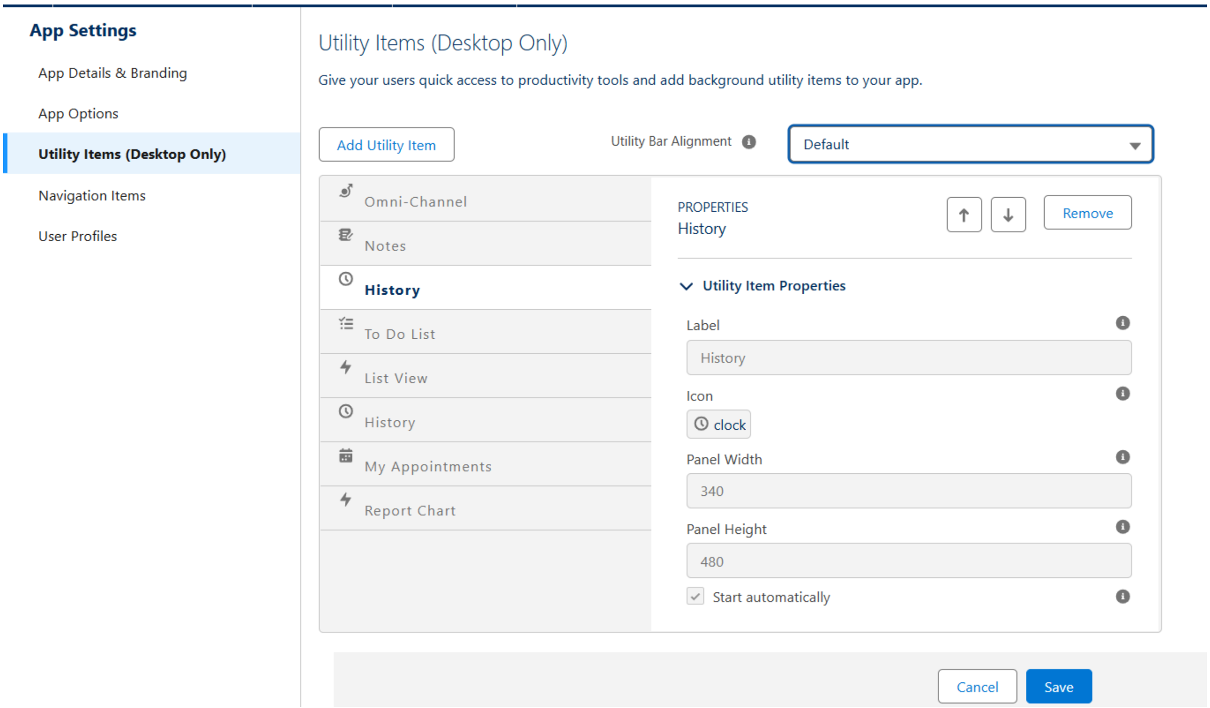
Service Console
(Lightning Experience) Lets support agents work with multiple...

Sales
Manage your sales process with accounts, leads, opportunities, and more

---

5. Utility Bar

- Persistent footer in Lightning apps.
- Quick access tools for productivity.
- Examples for Video project:
  - Video Uploader (LWC)
  - Recent Videos
  - Alerts from Detection System
- Add via **App Manager → Utility Bar → Add Utility**.



Utility Items (Desktop Only)

Give your users quick access to productivity tools and add background utility items to your app.

Utility Bar Alignment: Default

PROPERTIES
History

Utility Item Properties

Label: History

Icon: clock

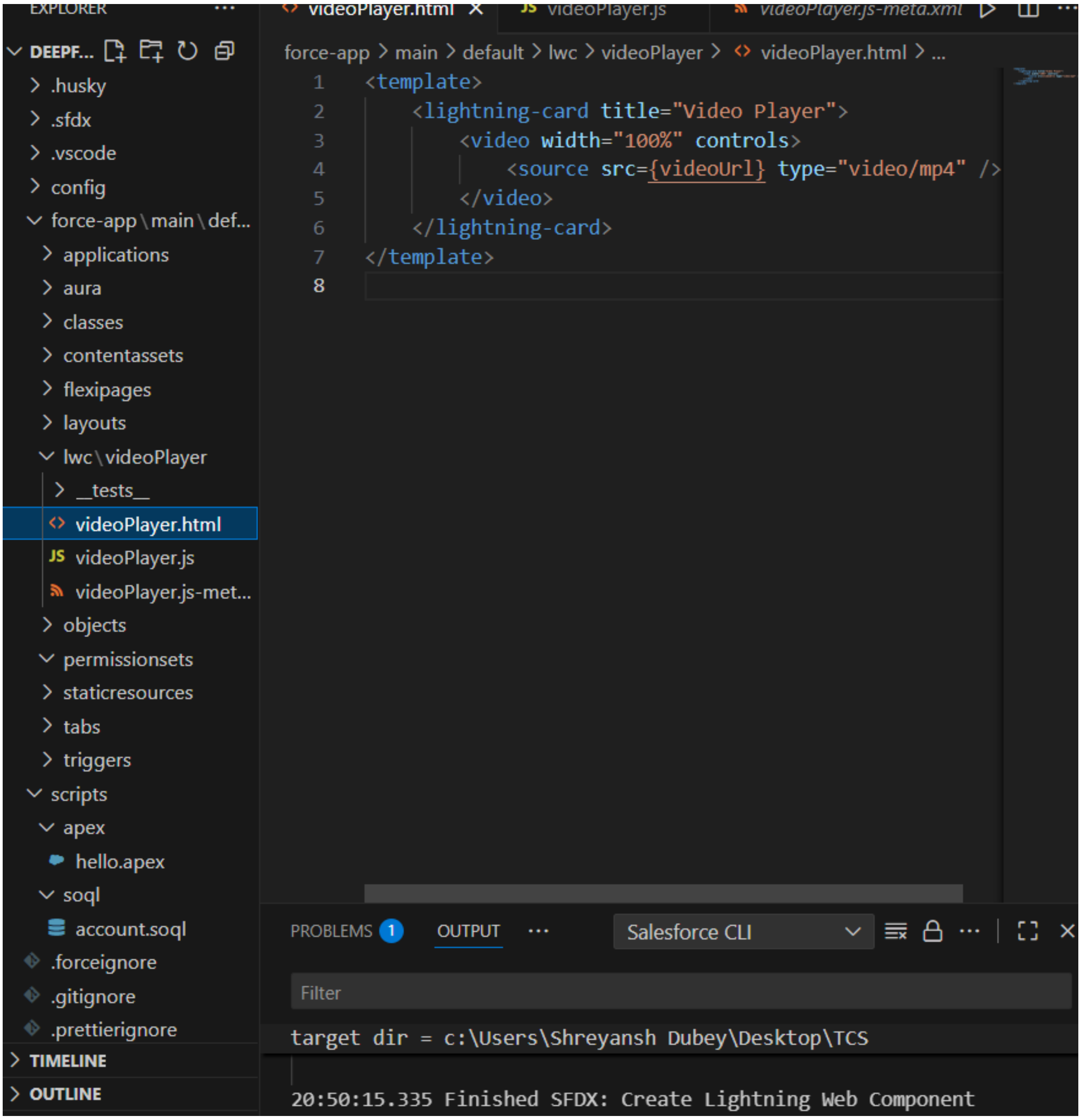
Panel Width: 340

Panel Height: 480

☑ Start automatically

6. Lightning Web Components (LWC)

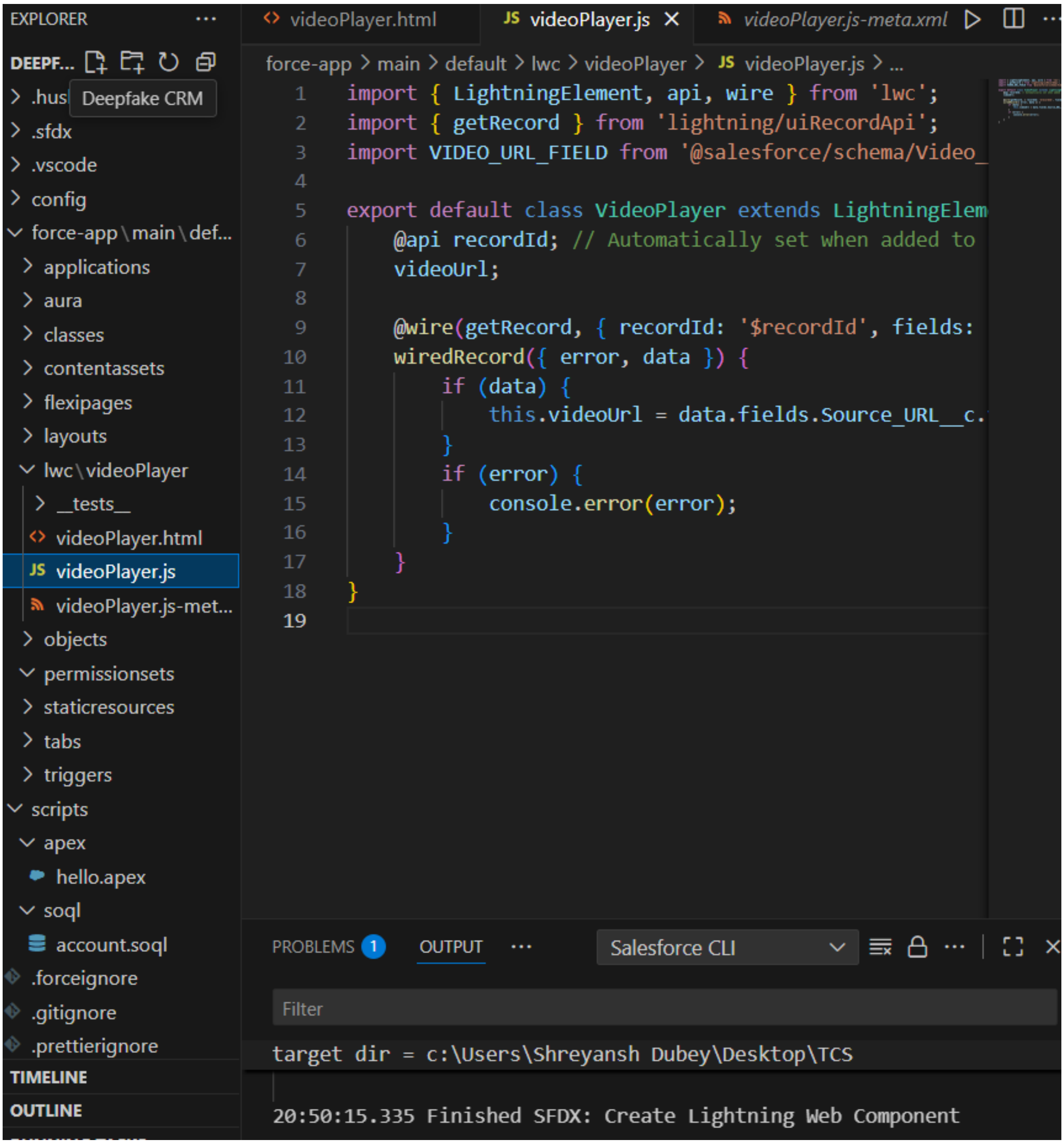
- Modern framework based on **JavaScript ES6+**.
- Files:
  - .html → UI
  - .js → Logic
  - .css → Styling (optional)
- Can be added to Record Pages, App Pages, and Utility Bar.
- Example: Video Player, Video Alerts, Upload Widget.



```html
<template>
    <lightning-card title="Video Player">
        <video width="100%" controls>
            <source src={videoUrl} type="video/mp4" />
        </video>
    </lightning-card>
</template>
```

```
target dir = c:\Users\Shreyansh Dubey\Desktop\TCS

20:50:15.335 Finished SFDX: Create Lightning Web Component
```



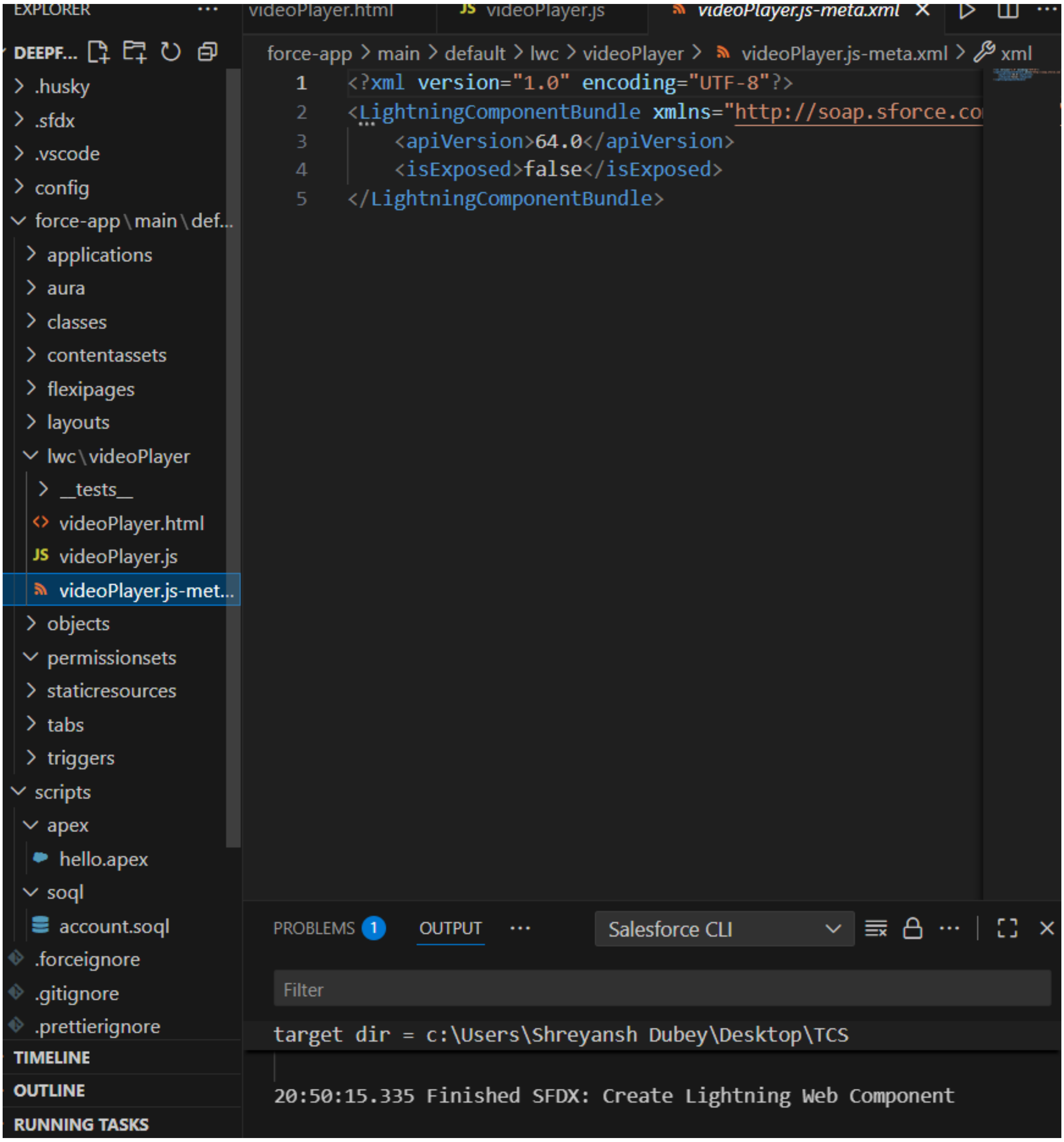
```
import { LightningElement, api, wire } from 'lwc';
import { getRecord } from 'lightning/uiRecordApi';
import VIDEO_URL_FIELD from '@salesforce/schema/Video_

export default class VideoPlayer extends LightningElem
    @api recordId; // Automatically set when added to
    videoUrl;

    @wire(getRecord, { recordId: '$recordId', fields:
    wiredRecord({ error, data }) {
        if (data) {
            this.videoUrl = data.fields.Source_URL__c.
        }
        if (error) {
            console.error(error);
        }
    }
}
```

```
target dir = c:\Users\Shreyansh Dubey\Desktop\TCS

20:50:15.335 Finished SFDX: Create Lightning Web Component
```



```
<?xml version="1.0" encoding="UTF-8"?>
<LightningComponentBundle xmlns="http://soap.sforce.co
    <apiVersion>64.0</apiVersion>
    <isExposed>false</isExposed>
</LightningComponentBundle>
```

```
target dir = c:\Users\Shreyansh Dubey\Desktop\TCS

20:50:15.335 Finished SFDX: Create Lightning Web Component
```

---

7. Apex with LWC

- Apex handles **server-side logic**.
- **Read-only data:** Use **@AuraEnabled(cacheable=true)** with wire adapter.
- **User actions:** Use **imperative Apex calls** for buttons/events.
- Example:
    - Fetch Video details

    - Update Video status
    - Analyze video

---

8. Events in LWC

- **Custom events:** Child → Parent communication.
- **Pub/Sub / Application events:** Component → unrelated component communication.
- Use case: Update Detection Status across multiple components when video is analyzed.

---

9. Wire Adapters

- Reactive data fetching.
- Examples:
    - getRecord
    - getObjectInfo
    - getPicklistValues
- Automatically updates component when data changes.

---

10. Imperative Apex Calls

- Explicit Apex call triggered by user action.
- Example:
    - Analyze video
    - Approve video
- Flexible control over when server-side code runs.

---

11. Navigation Service

- Use NavigationMixin in LWC.
- Programmatic navigation:
    - Standard pages (record, object list)

    - URLs
    - Lightning pages
- Example: Redirect user after saving or analyzing a video.
