

```

import { createSlice, createAsyncThunk } from "@reduxjs/toolkit";
import TraCuuService from "services/sale/TraCuuService";
import { cloneDeep } from "lodash";

export const getTraCuuTon = createAsyncThunk(
  "tracuu/getTraCuuTon",
  async (data, { rejectWithValue }) => {
    try {
      const response = await TraCuuService.getTraCuuTon(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const getTraCuuCongNo = createAsyncThunk(
  "tracuu/getTraCuuCongNo",
  async (data, { rejectWithValue }) => {
    try {
      const response = await TraCuuService.getTraCuuCongNo(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const getXemThongTinHanghoa = createAsyncThunk(
  "tracuu/getXemThongTinHanghoa",
  async (data, { rejectWithValue }) => {
    try {
      const response = await TraCuuService.getXemThongTinHanghoa(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const getDanhSachDaXuatHanghoa = createAsyncThunk(
  "tracuu/getDanhSachDaXuatHanghoa",
  async (data, { rejectWithValue }) => {
    try {
      const response = await TraCuuService.getDanhSachDaXuatHanghoa(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const capNhatTonHanghoa = createAsyncThunk(
  "tracuu/capNhatTonHanghoa",
  async (data, { rejectWithValue }) => {
    try {
      const { onSuccess } = data;
      const payload = cloneDeep(data);
      delete payload.onSuccess;
      const response = await TraCuuService.capNhatTonHanghoa(payload);
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

const initialState = {
  loading: false,
  tracuuTonList: [],
  tracuuCongNoList: [],
  xemthongtinhanghoaList: [],
  danhsachdaxuathanghoaList: [],
};

```

```

export const tracuuSlice = createSlice({
  name: "tracuu",
  initialState,
  reducers: {
    showLoading: (state) => {
      state.loading = true;
    },
    setAllDmkkList: (state, action) => {
      state.allDmKkList = action.payload;
    },
  },
},
extraReducers: (builder) => {
  builder
    .addCase(getTraCuuTon.pending, (state) => {
      state.loading = true;
    })
    .addCase(getTraCuuTon.fulfilled, (state, action) => {
      state.loading = false;
      state.tracuutonList = action.payload;
    })
    .addCase(getTraCuuTon.rejected, (state, action) => {
      state.loading = false;
    })
    .addCase(getTraCuuCongNo.fulfilled, (state, action) => {
      state.loading = false;
      state.tracuucongnoList = action.payload;
    })
    .addCase(getXemThongTinHanghoa.pending, (state) => {
      state.loading = true;
    })
    .addCase(getXemThongTinHanghoa.fulfilled, (state, action) => {
      state.loading = false;
      state.xemthongtinhanghoaList = action.payload;
    })
    .addCase(getXemThongTinHanghoa.rejected, (state, action) => {
      state.loading = false;
    })
    .addCase(getDanhSachDaXuatHanghoa.pending, (state) => {
      state.loading = true;
    })
    .addCase(getDanhSachDaXuatHanghoa.fulfilled, (state, action) => {
      state.loading = false;
      state.danh sachdaxuathanghoaList = action.payload;
    })
    .addCase(getDanhSachDaXuatHanghoa.rejected, (state, action) => {
      state.loading = false;
    })
    .addCase(capNhatTonHanghoa.pending, (state) => {
      state.loading = true;
    })
    .addCase(capNhatTonHanghoa.fulfilled, (state, action) => {
      state.loading = false;
    })
    .addCase(capNhatTonHanghoa.rejected, (state, action) => {
      state.loading = false;
    });
},
});

export const { showLoading } = tracuuSlice.actions;

export default tracuuSlice.reducer;

```