

```

import { createSlice, createAsyncThunk } from "@reduxjs/toolkit";
import ThietLapLuongService from "services/salary/ThietLapLuongService.js";

export const thietlapluongGetAll = createAsyncThunk(
  "thietlapluong/thietlapluongGetAll",
  async (data, { rejectWithValue }) => {
    try {
      const response = await ThietLapLuongService.getAll(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const thietLapPhuCapGiamTruGetAll = createAsyncThunk(
  "thietlapluong/thietLapPhuCapGiamTruGetAll",
  async (data, { rejectWithValue }) => {
    try {
      const response = await ThietLapLuongService.thietlapphucapgiamtruGetAll(
        data
      );
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const createThietLapPhuCapGiamTru = createAsyncThunk(
  "thietlapluong/createThietLapPhuCapGiamTru",
  async (data, { rejectWithValue }) => {
    try {
      const { onSuccess } = data;
      const response = await ThietLapLuongService.createThietLapPhuCapGiamTru(
        data
      );
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const updateThietLapPhuCapGiamTru = createAsyncThunk(
  "thietlapluong/updateThietLapPhuCapGiamTru",
  async (data, { rejectWithValue }) => {
    const { onSuccess } = data;
    try {
      const response = await ThietLapLuongService.updateThietLapPhuCapGiamTru(
        data
      );
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

export const deleteThietLapPhuCapGiamTru = createAsyncThunk(
  "thietlapluong/deleteThietLapPhuCapGiamTru",
  async (data, { rejectWithValue }) => {
    const { onSuccess, id } = data;
    try {
      const response = await ThietLapLuongService.deleteThietLapPhuCapGiamTru(
        id
      );
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);

```

```

);
const initialState = {
  loading: false,
  thietlapluong: {},
  thietlapphucapgiamtruList: [],
};

export const thietlapluongSlice = createSlice({
  name: "thietlapluong",
  initialState,
  reducers: {
    showLoading: (state) => {
      state.loading = true;
    },
  },
  extraReducers: (builder) => {
    builder
      .addCase(thietlapluongGetAll.pending, (state) => {
        state.loading = true;
      })
      .addCase(thietlapluongGetAll.fulfilled, (state, action) => {
        state.loading = false;
        state.thietlapluong = action.payload;
      })
      .addCase(thietlapluongGetAll.rejected, (state, action) => {
        state.loading = false;
      })
      .addCase(thietLapPhuCapGiamTruGetAll.pending, (state) => {
        state.loading = true;
      })
      .addCase(thietLapPhuCapGiamTruGetAll.fulfilled, (state, action) => {
        state.loading = false;
        state.thietlapphucapgiamtruList = action.payload;
      })
      .addCase(thietLapPhuCapGiamTruGetAll.rejected, (state, action) => {
        state.loading = false;
      });
  },
});

export const { showLoading } = thietlapluongSlice.actions;

export default thietlapluongSlice.reducer;

```