

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

```
export const getAllDichVuTaiNhaCategory = createAsyncThunk(
  "dichvutainhacategory/getAllDichVuTaiNhaCategory",
  async (data, { rejectWithValue }) => {
    try {
      const response = await DichVuTaiNhaCategoryService.searchGrid(data);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const getDichVuTaiNhaCategoryById = createAsyncThunk(
  "dichvutainhacategory/getDichVuTaiNhaCategoryById",
  async (id, { rejectWithValue }) => {
    try {
      const response =
        await DichVuTaiNhaCategoryService.getDichVuTaiNhaCategoryById(id);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const createDichVuTaiNhaCategory = createAsyncThunk(
  "dichvutainhacategory/createDichVuTaiNhaCategory",
  async (data, { rejectWithValue }) => {
    try {
      const { onSuccess } = data;
      const payload = cloneDeep(data);
      delete payload.onSuccess;
      const response = await DichVuTaiNhaCategoryService.create(payload);
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const updateDichVuTaiNhaCategory = createAsyncThunk(
  "dichvutainhacategory/updateDichVuTaiNhaCategory",
  async (data, { rejectWithValue }) => {
    try {
      const { onSuccess } = data;
      const payload = cloneDeep(data);
      delete payload.onSuccess;
      const response = await DichVuTaiNhaCategoryService.update(payload);
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```
export const deletedDichVuTaiNhaCategory = createAsyncThunk(
  "dichvutainhacategory/deletedDichVuTaiNhaCategoryById",
  async (data, { rejectWithValue }) => {
    try {
      const { onSuccess, id } = data;
      const response = await DichVuTaiNhaCategoryService.delete(id);
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
```

```

    }
  }
);
export const changeStateDichVuTaiNhaCategory = createAsyncThunk(
  "dichvutainhacategory/changeStateDichVuTaiNhaCategory",
  async (data, { rejectWithValue }) => {
    try {
      const { onSuccess } = data;
      const payload = cloneDeep(data);
      delete payload.onSuccess;
      const response = await DichVuTaiNhaCategoryService.changeStateHospital(
        payload
      );
      if (onSuccess) onSuccess(response);
      return response.data;
    } catch (err) {
      return rejectWithValue(err.message || "Error");
    }
  }
);
const initialState = {
  loading: false,
  dichVuTaiNhaCategoryList: [],
  dichVuTaiNhaCategoryDetail: {},
};

export const dichVuTaiNhaCategorySlice = createSlice({
  name: "dichvutainhacategory",
  initialState,
  reducers: {
    showLoading: (state) => {
      state.loading = true;
    },
    setHisInfo: (state, action) => {
      state.hisInfoList = action.payload;
    },
  },
  extraReducers: (builder) => {
    builder
      .addCase(getAllDichVuTaiNhaCategory.pending, (state) => {
        state.loading = true;
      })
      .addCase(getAllDichVuTaiNhaCategory.fulfilled, (state, action) => {
        state.loading = false;
        state.dichVuTaiNhaCategoryList = action.payload;
      })
      .addCase(getAllDichVuTaiNhaCategory.rejected, (state, action) => {
        state.loading = false;
      })
      .addCase(getDichVuTaiNhaCategoryById.pending, (state) => {
        state.loading = true;
      })
      .addCase(getDichVuTaiNhaCategoryById.fulfilled, (state, action) => {
        state.loading = false;
        state.dichVuTaiNhaCategoryDetail = action.payload;
      })
      .addCase(getDichVuTaiNhaCategoryById.rejected, (state, action) => {
        state.loading = false;
      })
      .addCase(createDichVuTaiNhaCategory.pending, (state) => {
        state.loading = true;
      })
      .addCase(createDichVuTaiNhaCategory.fulfilled, (state, action) => {
        state.loading = false;
      })
      .addCase(createDichVuTaiNhaCategory.rejected, (state, action) => {
        state.loading = false;
      })
      .addCase(updateDichVuTaiNhaCategory.pending, (state) => {
        state.loading = true;
      })
      .addCase(updateDichVuTaiNhaCategory.fulfilled, (state, action) => {
        state.loading = false;
      })
  }
});

```

```

        .addCase(updatedDichVuTaiNhaCategory.rejected, (state, action) => {
            state.loading = false;
        })
        .addCase(deletedDichVuTaiNhaCategory.pending, (state) => {
            state.loading = true;
        })
        .addCase(deletedDichVuTaiNhaCategory.fulfilled, (state, action) => {
            state.loading = false;
        })
        .addCase(deletedDichVuTaiNhaCategory.rejected, (state, action) => {
            state.loading = false;
        })
        .addCase(changeStateDichVuTaiNhaCategory.pending, (state) => {
            state.loading = true;
        })
        .addCase(changeStateDichVuTaiNhaCategory.fulfilled, (state, action) => {
            state.loading = false;
        })
        .addCase(changeStateDichVuTaiNhaCategory.rejected, (state, action) => {
            state.loading = false;
        });
    },
});

export const { showLoading, setHisInfo } = dichVuTaiNhaCategorySlice.actions;

export default dichVuTaiNhaCategorySlice.reducer;

```