cmake\_minimum\_required(VERSION 3.5.1)

project(grid\_map\_visualization)

set(CMAKE\_CXX\_FLAGS "-std=c++11 ${CMAKE\_CXX\_FLAGS}")

add\_compile\_options(-Wall -Wextra -Wpedantic)

set(CMAKE\_EXPORT\_COMPILE\_COMMANDS ON)

## Find catkin macros and libraries

find\_package(catkin REQUIRED COMPONENTS

roscpp

tf

grid\_map\_core

grid\_map\_ros

grid\_map\_msgs

visualization\_msgs

sensor\_msgs

nav\_msgs

)

## System dependencies are found with CMake's conventions

#find\_package(Eigen3 REQUIRED)

###################################

## catkin specific configuration ##

###################################

## The catkin\_package macro generates cmake config files for your package

## Declare things to be passed to dependent projects

## INCLUDE\_DIRS: uncomment this if you package contains header files

## LIBRARIES: libraries you create in this project that dependent projects also need

## CATKIN\_DEPENDS: catkin\_packages dependent projects also need

## DEPENDS: system dependencies of this project that dependent projects also need

catkin\_package(

INCLUDE\_DIRS

include

# LIBRARIES ${PROJECT\_NAME}

CATKIN\_DEPENDS

roscpp

tf

grid\_map\_core

grid\_map\_ros

grid\_map\_msgs

visualization\_msgs

sensor\_msgs

nav\_msgs

)

###########

## Build ##

###########

## Specify additional locations of header files

## Your package locations should be listed before other locations

include\_directories(

include

SYSTEM

${catkin\_INCLUDE\_DIRS}

${EIGEN3\_INCLUDE\_DIR}

)

## Declare a cpp library

# add\_library(${PROJECT\_NAME}

# src/${PROJECT\_NAME}/grid\_map\_visualization\_node.cpp

# )

## Declare a cpp executable

add\_executable(${PROJECT\_NAME}

src/grid\_map\_visualization\_node.cpp

src/GridMapVisualization.cpp

src/GridMapVisualizationHelpers.cpp

src/visualizations/VisualizationBase.cpp

src/visualizations/VisualizationFactory.cpp

src/visualizations/PointCloudVisualization.cpp

src/visualizations/FlatPointCloudVisualization.cpp

src/visualizations/VectorVisualization.cpp

src/visualizations/OccupancyGridVisualization.cpp

src/visualizations/GridCellsVisualization.cpp

src/visualizations/MapRegionVisualization.cpp

)

## Add cmake target dependencies of the executable/library

## as an example, message headers may need to be generated before nodes

# add\_dependencies(grid\_map\_visualization\_node grid\_map\_visualization\_generate\_messages\_cpp)

## Specify libraries to link a library or executable target against

target\_link\_libraries(

${PROJECT\_NAME}

${catkin\_LIBRARIES}

)

#############

## Install ##

#############

# Mark executables and/or libraries for installation

install(

TARGETS ${PROJECT\_NAME}

ARCHIVE DESTINATION ${CATKIN\_PACKAGE\_LIB\_DESTINATION}

LIBRARY DESTINATION ${CATKIN\_PACKAGE\_LIB\_DESTINATION}

RUNTIME DESTINATION ${CATKIN\_PACKAGE\_BIN\_DESTINATION}

)

# Mark other files for installation

install(

DIRECTORY doc

DESTINATION ${CATKIN\_PACKAGE\_SHARE\_DESTINATION}

)

#############

## Testing ##

#############

if (CATKIN\_ENABLE\_TESTING)

catkin\_add\_gtest(${PROJECT\_NAME}-test

test/test\_grid\_map\_visualization.cpp

test/empty\_test.cpp

)

add\_dependencies(${PROJECT\_NAME}-test

${PROJECT\_NAME}

)

target\_include\_directories(${PROJECT\_NAME}-test PRIVATE

include

)

target\_include\_directories(${PROJECT\_NAME}-test SYSTEM PUBLIC

${catkin\_INCLUDE\_DIRS}

${EIGEN3\_INCLUDE\_DIRS}

)

###################

## Code\_coverage ##

###################

find\_package(cmake\_code\_coverage QUIET)

if(cmake\_code\_coverage\_FOUND)

add\_gtest\_coverage(

TEST\_BUILD\_TARGETS

${PROJECT\_NAME}-test

)

endif()

endif()

#################

## Clang\_tools ##

#################

find\_package(cmake\_clang\_tools QUIET)

if(cmake\_clang\_tools\_FOUND)

add\_default\_clang\_tooling(

DISABLE\_CLANG\_FORMAT

)

endif(cmake\_clang\_tools\_FOUND)