**Dissertation**

**Title:**

**Topological Navigation Editor**

**(ToNE)**

By

Laxwell Nyama

MComp Computer Science

University of Lincoln

2015/2016

Supervised by

Dr Marc Hanheide

1. **Abstract**

Graphic user interfaces have grown in popularity and use over the past few years with almost over three quarters of the world using them. This project seeks to develop an innovative graphical user interface to be used in robotics for topological navigation edition. The intention is to designing a topological navigation map editor interface (GUI), which is robust, functional, usable and accessible, to be used by people with no technical knowledge or background, to control mobile robots. The report will then discuss the aims and objectives decided after the literature review before moving onto the method and implementation details of the artifact. Finally there will be an evaluation section which

1. **Acknowledgement**

A long and ardours journey it has been, towards the attainment of my BSc (Hons). It would not have been possible without the help of all those I came across at various stage of my project. I would like to extend my immense gratitude to all those who lent me their support in diverse ways during the trials and tribulations I encountered in the course of this project, those who helped me through cold winters, those who brought a smile to my face, gave me advice, motivation and encouragement to do the best of my ability.

First and foremost I would like to thank my supervisor, Dr Marc Hanheide. His mentorship, perseverance and willingness to help made it more bearable. His advices, guidance and support encouraged me to evolve from a struggling student into an ambitious researcher. He taught me how to effectively use communication tools available to improve collaboration. Thanks to many hours of discussion with him, I learnt how to undertake a research project. His aim for perfection helped me improve my research process and polish my skill-set.

I also extend my gratitude to J who is my co-supervisor for his contributions to the success of my project. As a pair-programming partner, I learnt a lot from him regarding the problem domain and most.

It would not be fair not to mention my friend James Chadwick, the man with the Midas touch, who has been helping with most of all the problems I encountered. Also as my pair-programming partner on many occasions, he helped me maintain high quality code and good coding practices. His involvement is greatly appreciated.

Lastly I would like to thank my family for the love, support and faith they showed in me, not to forget their encouraging words of wisdom.

**Table of Contents**

**Page**

1. Introduction 3
2. Aim and Objectives
   1. Aims 4
   2. Objectives 4
   3. Motivation 8
3. Requirements
4. Background Research 9
   1. Overview 9
   2. Topological Navigation 9
   3. Autonomous mobile robots and Navigation 9
   4. Simulation Agents 10
   5. Graphical User Interface 10
5. Design 14
   1. Overview 14
   2. Design 14

14

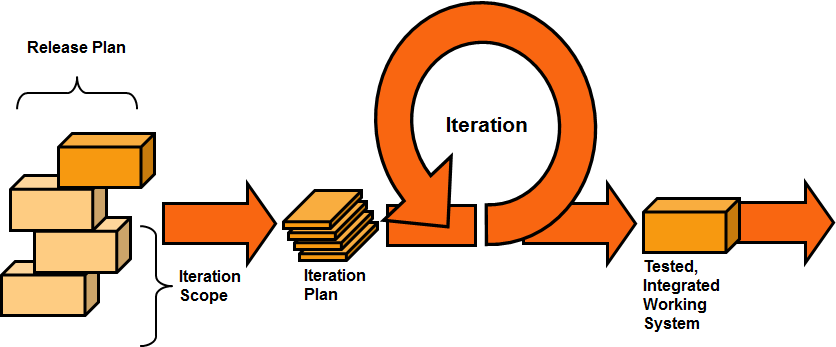
1. Implementation
2. Evaluation
3. Resources 18
   1. Overview 18
   2. Development Platform 18
   3. Hardware requirements 18
   4. Software requirements 19
4. Conclusion 20
5. References 21
6. **Introduction**

“Autonomous Mobile robotics has seen exponential growth in recent years and robot navigation is a growing area of scientific research. Without navigation the creation of self-propelled, household machines, guard robots, or planet surveyors is beyond imagination”. (Richárd Szabó, 2004). An easy and effective way of controlling such robots will vastly increase their applicability. The exponential growth in mobile robotics and robot use has called for easier and simpler ways to interact and control them, however, the control and use of these robots requires a high level of technical knowledge. The way these robots are currently controlled is too complex and technical for an average person, hence the need to design and implement a dedicated graphical user interface that is simple, effective and easy to use for people with little to no technical ability.

This project is part of STRANDS project; STRANDS aims to enable a robot to achieve robust and intelligent behaviour in human environments through exploitation of long-term experience in security and care scenarios. This project aim to develop a real world topological navigation editor interface (GUI) by improving on the existing techniques to advance the robot navigation capabilities, usability and user interfaces. The development of robust and effective navigation editor that can be used by people with no technical background will play a great part in the deployment of these robots. Autonomous Mobile Robotics is a growing scientific field and it is one of the modules I will be undertaking.

The implementation of the interface is carried out using an agile software development methodology known as Extreme Programming (XP), PyQt4 and Python programming language.

Extreme Programming (XP) is a software engineering methodology, the most prominent of several agile software development methodologies. Like other agile methodologies, Extreme Programming differs from traditional methodologies primarily in placing a higher value on adaptability than on predictability. “XP also focuses on excellent application of programming techniques, clear communication, feedback, courage and respect, which allows us to accomplish things we previously could not even imagine”(Kent Beck and Cynthia Andres, 2005). These are the reasons for choosing to use this methodology and below is a figure illustrating XP Programming:



The language of choice for developing the artifact is python as this is a requirement for systems compatibility purposes. “Python is a simple but powerful object-orientated language. Its simplicity makes it easy to learn, but its power means that large and complex applications can be created. Its interpreted nature means that Python programmers are very productive because there is no edit/compile/link/run development cycle.” (The Qt Company Ltd, 2015).

Also as a requirement, PyQt4 is the tool used to develop the graphic user interface. PyQt is a set of Python v2 and v3 bindings for the cross-platform Qt application framework developed in C++ for designing graphical user interfaces, which runs on all platforms supported by Qt including Windows, MacOS/X and Linux. PyQt brings together the Qt C++ cross-platform application framework and the cross-platform interpreted language Python. (The Qt Company Ltd, 2015).

1. **Aims and objectives**

**Aim**: The aim of the project is to develop a dedicated Topological navigation editor user interface for robot systems that can be used by people with no technical background.

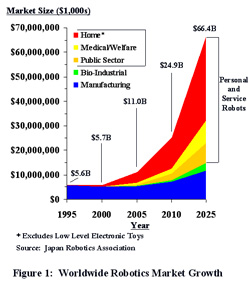
**Objectives**

* Gather requirements
* Project proposal
* To identify, investigate and evaluate current systems (Topological navigation editors)
* To identify, investigate and evaluate tools and techniques
* Create and evaluate a prototype
* Implement the topological navigation editor user interface
* Test, analyse and evaluate the topological navigation editor
* Demonstrate the effectiveness of the artefact
* Artefact to be delivered on time

**Motivation**

The two primary sources of motivation for this project came from the practical applications of autonomous mobile robots, and the more general insight to be gained in the field of robotics. However the main motivation is the current application of autonomous systems. In this day and age, everything is becoming autonomous, from household goods like hoovers, to factory machines and even automobiles. The figure below helps visualise the growth and expected growth in robotics use in different sectors in terms of market size.

Figure 1: Worldwide Robotics Market Growth



\**Excludes low level electronic toys*

*Source: Japan Robotics Association*

The multi-colored areas of the graph that rise above the blue area depict expected growth in *mobile* robotics. The blue area depicts maturation in the fixed robotics area.

In his article “The Robots Are Here” (Feb 2004 *MIT Technology Review*), Dr. Rodney Brooks, Director of MIT’s Computer Science and Artificial Intelligence Lab, commented, “I am convinced that robots are today where computers were in 1978. That’s about the year that computers started to appear around us in the way that robots are cropping up today. Of course, it was another 15 years before computers truly became pervasive in our lives. I think that 15 years from now, robots will be everywhere, as email and the Web are now.”

This project will also add greatly to the field of robotics in general, as it is still a growing field with a lot of research still to be undertaken where it pertains the edition of topological navigators.

1. **User Requirements**

The user requirements involve the creation of tools that will simplify the map creation/edition process making the whole system more user friendly. Simplify the use of the tools already available for map creation such as RVIz, simultaneous metric and topological map creation and direct map edition from text files. Improve and complete the tools for editing previously created maps using Rviz. Create a simple dedicated graphical user interface for editing the topological map.

The navigation editor design is to be simple for ease of control of the robots by people of all walks of like with little to no technical ability. The editor will include features like drag and drop, the user will be able to drag and drop a node, edge and also edit these through their properties tab by clicking on the node or edge itself. The user should also be able to edit maps created by other applications like Rviz and also be able to edit maps directly from text files. This implies that the user be able to upload maps for editing. Editing of the maps will include but not limited to adding, deleting, moving, naming nodes and edges,

However, while the graphical interface is aimed at user with no technical knowledge, expert users will also benefit.

“In order to design a system, it is necessary first to understand what it is that the system should be doing. The process of understanding the job to be done by the system is provided by task analysis”(Christine Faulkner 1998). In order to effectively design an effective system I need to understand the requirements (user and system). (Preece et al, 1994) goes on to say “understanding requirements involves looking at similar products, discussing the needs of the people who will use the product, and analysing any existing system to discover problems with current designs.” In my design I will use different principles, guidelines and methodologies to achieve the most effective design.

The objectives will be divided in terms of usability goals and user experience goals. According to (Preece et al, 2011) usability refers to ensuring that interactive products are easy to learn, effective to use and enjoyable from the users perspective. With this in mind my design will also try to cater for those students with various impairments so they can also have an enjoyable experience using the application

Usability goals provide the interaction designer with a concrete means of assessing various aspects of an interactive product and user experience, and just to reiterate this approach will be very effective hence my decision to implement it. “More specifically, usability can be broken into the following goals:

* Effectiveness
* Efficiency
* Safety
* Utility
* Learnability
* Memorability

A diversity of user experience goals has been articulated in interaction design, which covers a range of emotions and felt experiences. These include desirable and undesirable aspects:

Desirables:

* Cognitively stimulating
* Enjoyable
* Exciting
* Helpful
* Motivating
* Satisfying ” (Preece et al, 2011)

However there are a few constraints that will need to be taken into consideration when designing the application and these includes things like the cost, technical faults, convenience etc. The main point in this section is the gathering of requirements so as to establish the basis of the design. The application will need to be able to perform the intended task such as editing nodes, editing edges and drag and drop edges and nodes that are simple and intuitive, but the client from the requirements gathering will decide how this information should be presented.

Building upon previous research on how autonomous mobile robotics technology is employed in different facets of life and how the interfaces are designed I have managed to come up with some ideas of what functions to include and how to design a user centred design.

Using eXtreme Programming methodology, requirements are gathered through user stories. Below is a list of most of the features expected on the user interface, however some of these are disposable in accordance to the user needs:

* Edit nodes - add, delete, drag and drop, name, colour, size
* Edit edges - add, delete, drag and drop, name, colour, size
* Edit Maps - upload, drag and drop nodes and edges, rename, save.
* Auto-saving
* Feedback
* Robot position

Below are the use case diagrams that depict the user stories from the requirements gathering exercise.

**Use case diagrams**

Editing nodes and edges

Editing maps

1. **Background research**

Overview

The background research is fundamental in undertaking this project. It will furnish the

Topological Navigation

Ros , rviz

Autonomous mobile robots and Navigation

Simulation Agents

Graphical User Interface

PyQt4

GUI (Graphical User Interface) is a type of interface that allows users to communicate

1. **Design**

Overview

Façade design pattern

UML Diagram

User interface design requires properly identifying user needs and expectations. The ISO 9241-11 *Ergonomics of Human System Interaction* provides guidance on usability that is defined as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" (ISO 9241-11, 1998). Considering these three criteria, a prototype of the user interface will be made first and submitted to the client for evaluation. The feedback will then be used to design the ultimate interface design. User interface design is an iterative process where users interact with designers and interface prototypes to decide on features organisation and look and feel of the system user interface (Sommerville, 2007)

Design process needs to take into account basic GUI functionalities like opening and saving files, designing menus and widgets as well as HCI issues including cognitive load and cost of action for users. The GUI should provide capability of editing maps; this capability is gained by implementing add node and edge, drag and drop, mirroring facilities.

**Low-Fidelity Prototype**

“A Low-fidelity prototype also known as Lo-fi is one that does not look very much like the final product, it uses materials that are very different from the intended final version, such as paper and cardboard rather than electronic screens and metal.” (Preece et al, 2011). Paper prototyping involves use of paper, index cards, cardboard etc. to build up a paper representation of the system. One example of this was the “cardboard computer” (Ehn and Kyng, 1991).

Lo-fi prototyping is used for the benefits it offers such as being cheap and quick to produce. Lo-fi prototypes are also simple, cheap and quick to modify hence they support the exploration of alternative designs and ideas. This is particularly important during the user interface conceptual design.

First Iteration

During the first phase of the interface design using paper prototyping

Second iteration

**Implementation**

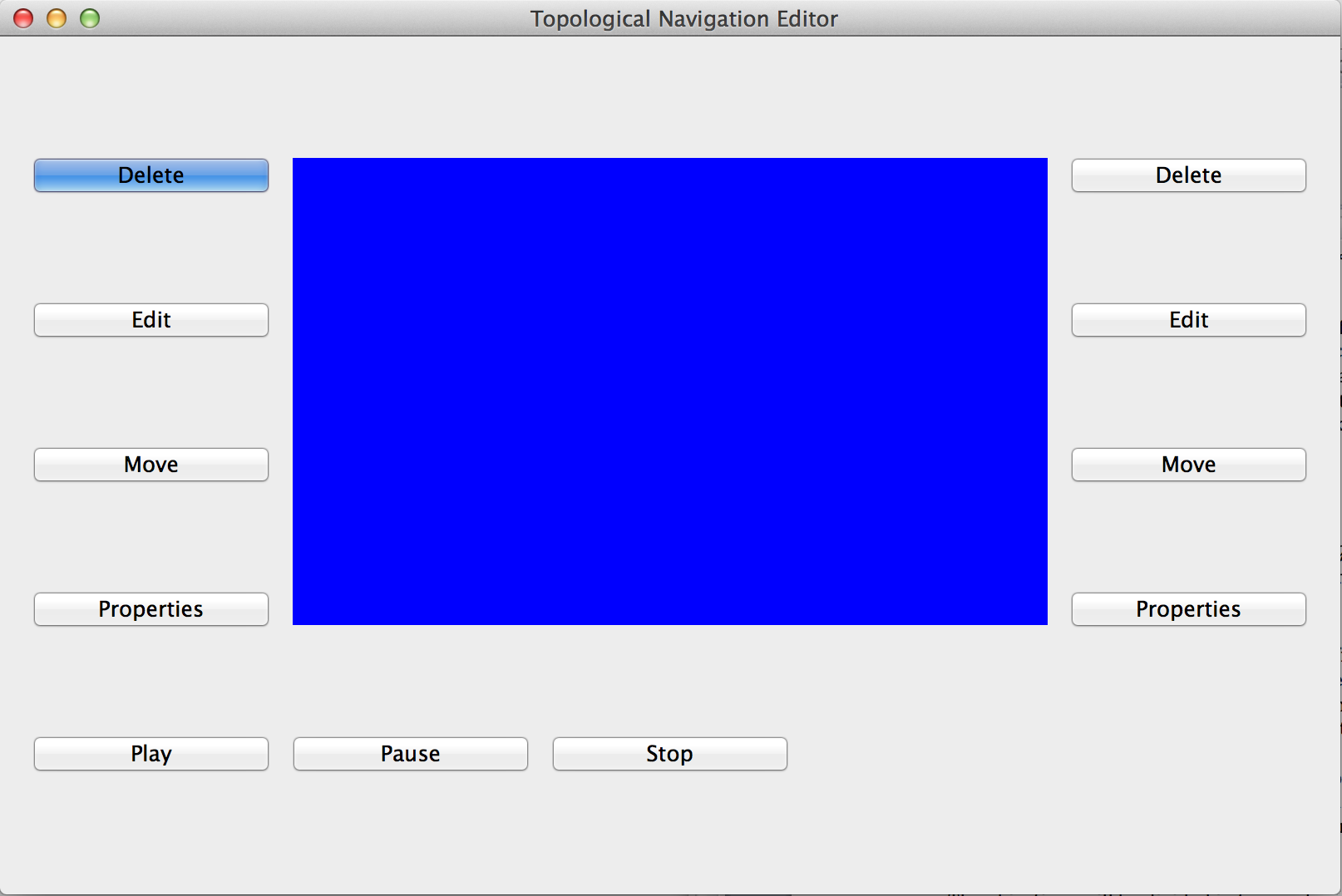
Methodology

**First Iteration**

Main window and layout

Wireframe of the user interface

The first iteration meant was just the implementation of the main window layout, which is shown by the figure below:

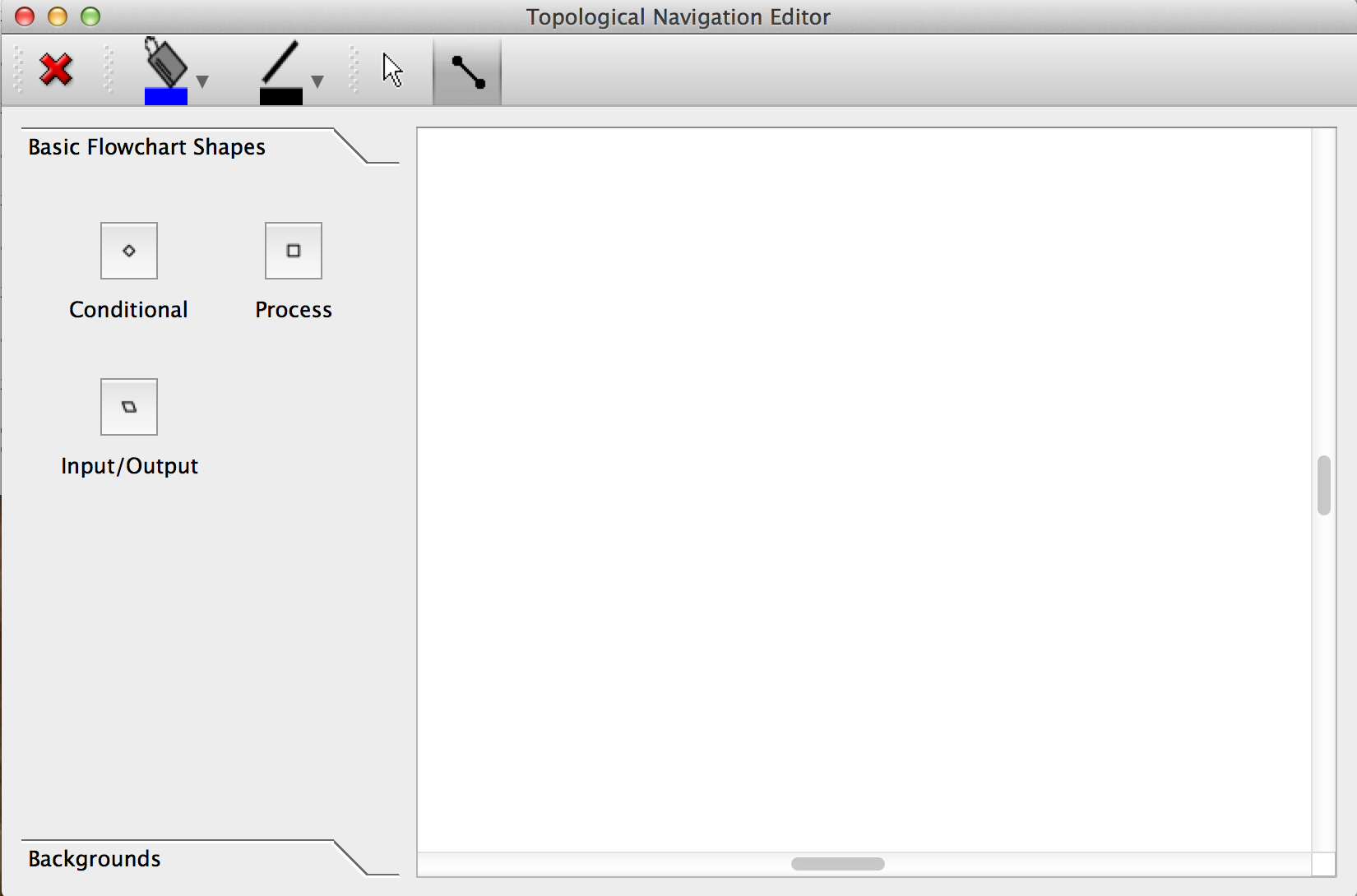


A grid layout has been used to position the toolbox, scene and the property window. This was achieved by using adding widgets in each layout

**Second Iteration**

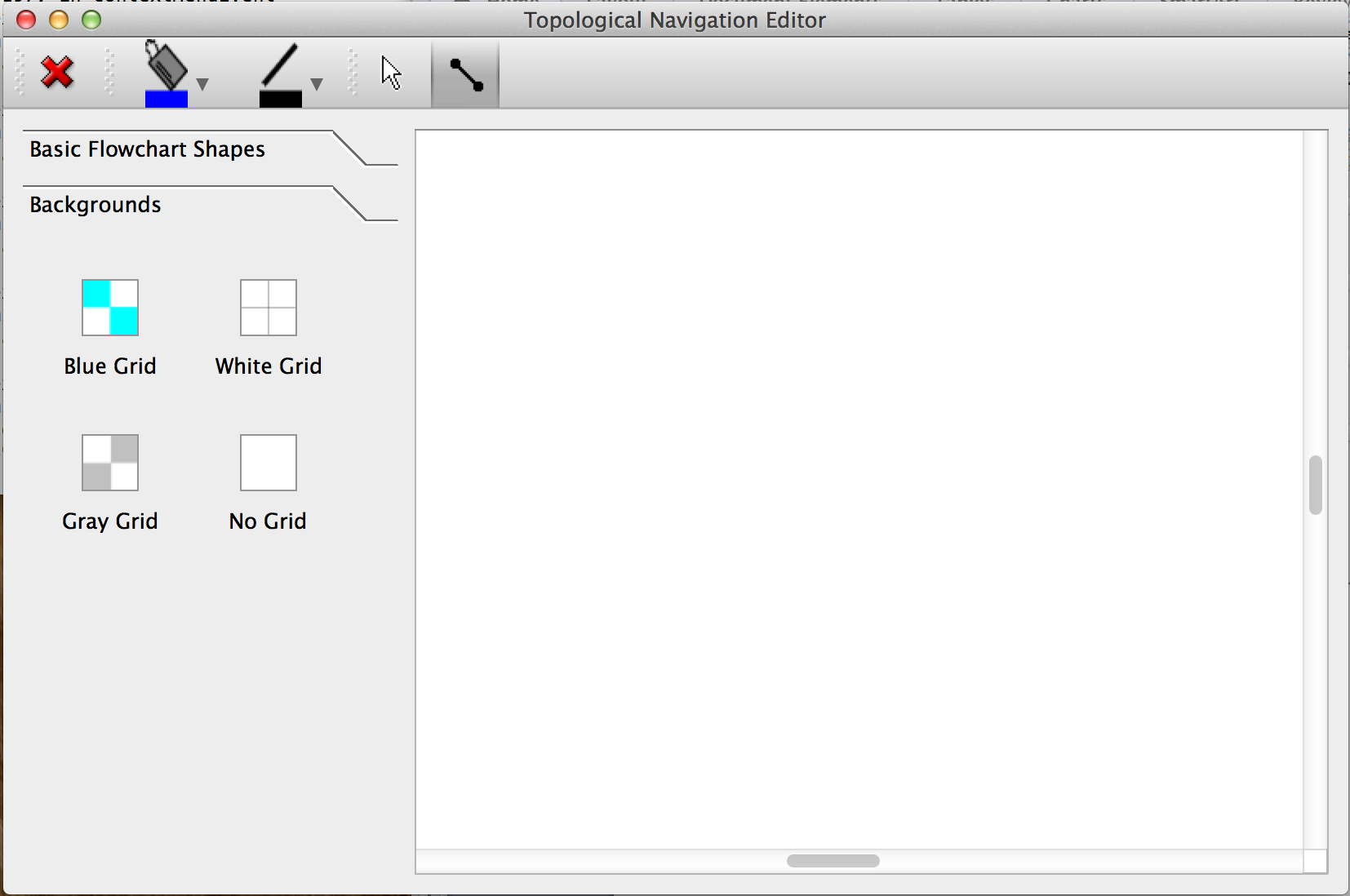
Toolbar and toolbox

The second phase involved the implementation of the toolbar and toolbox. The toolbox includes the nodes, background and map loading buttons.



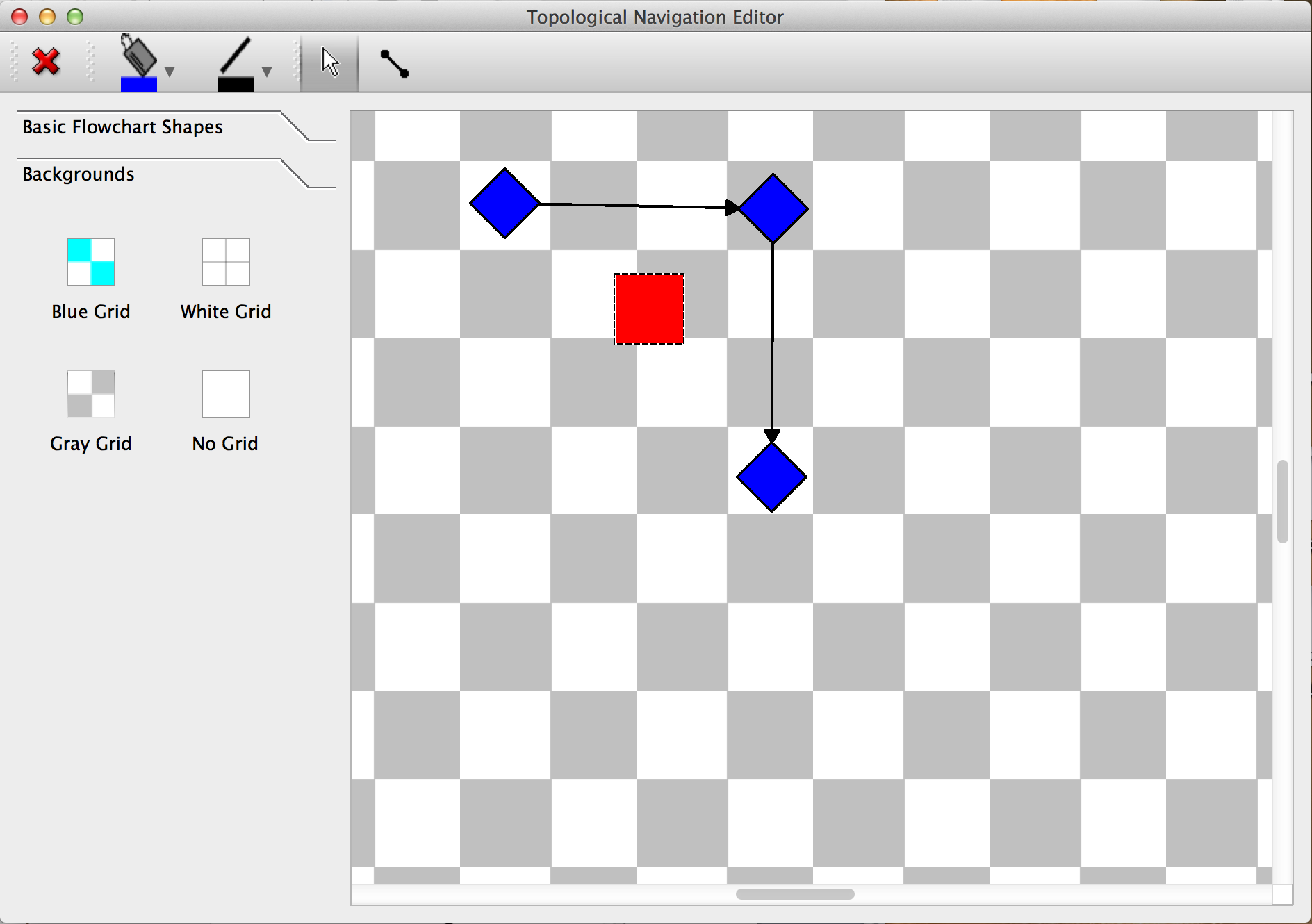
**Third Iteration**

Edges



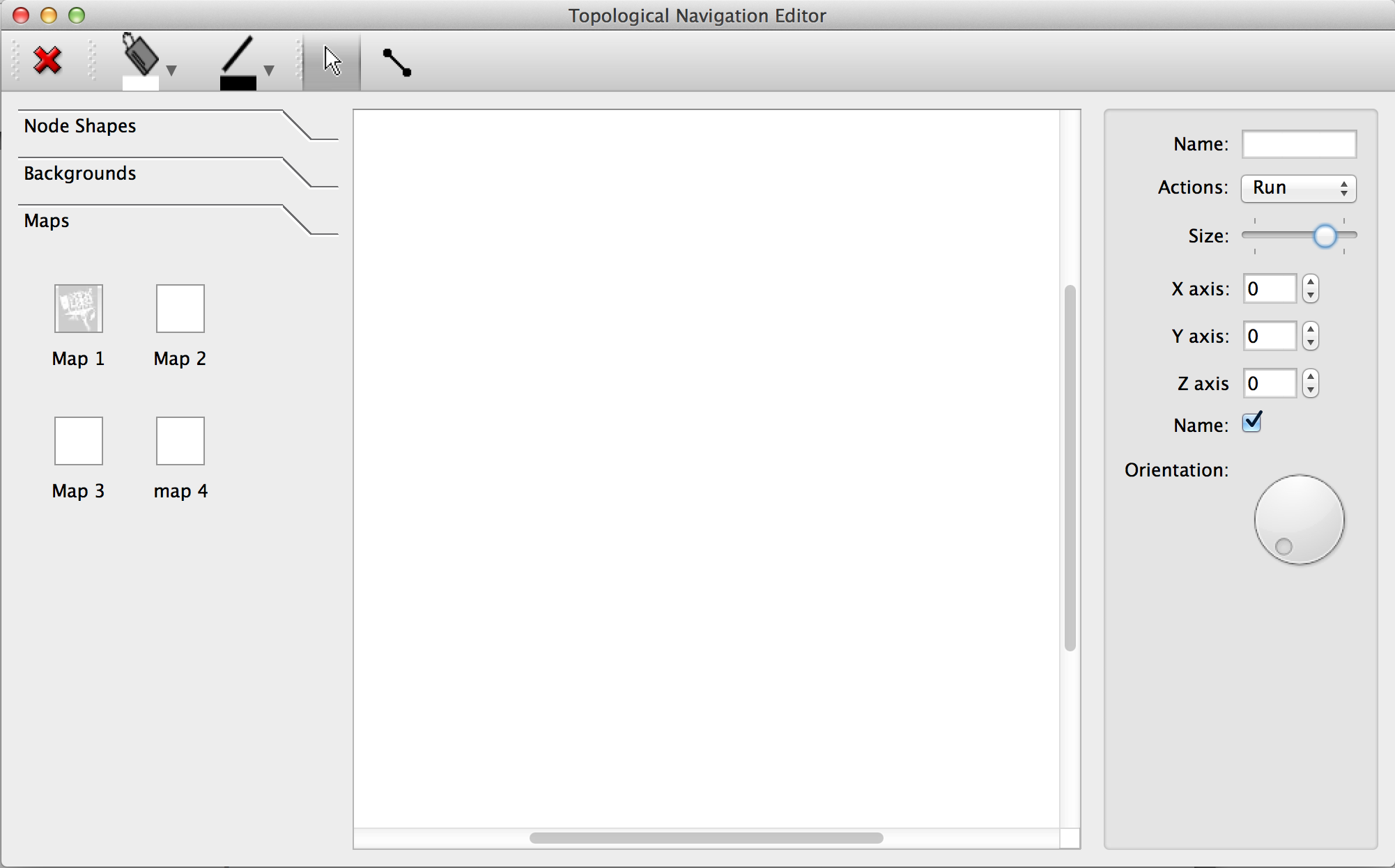
**Fourth Iteration**

Menus



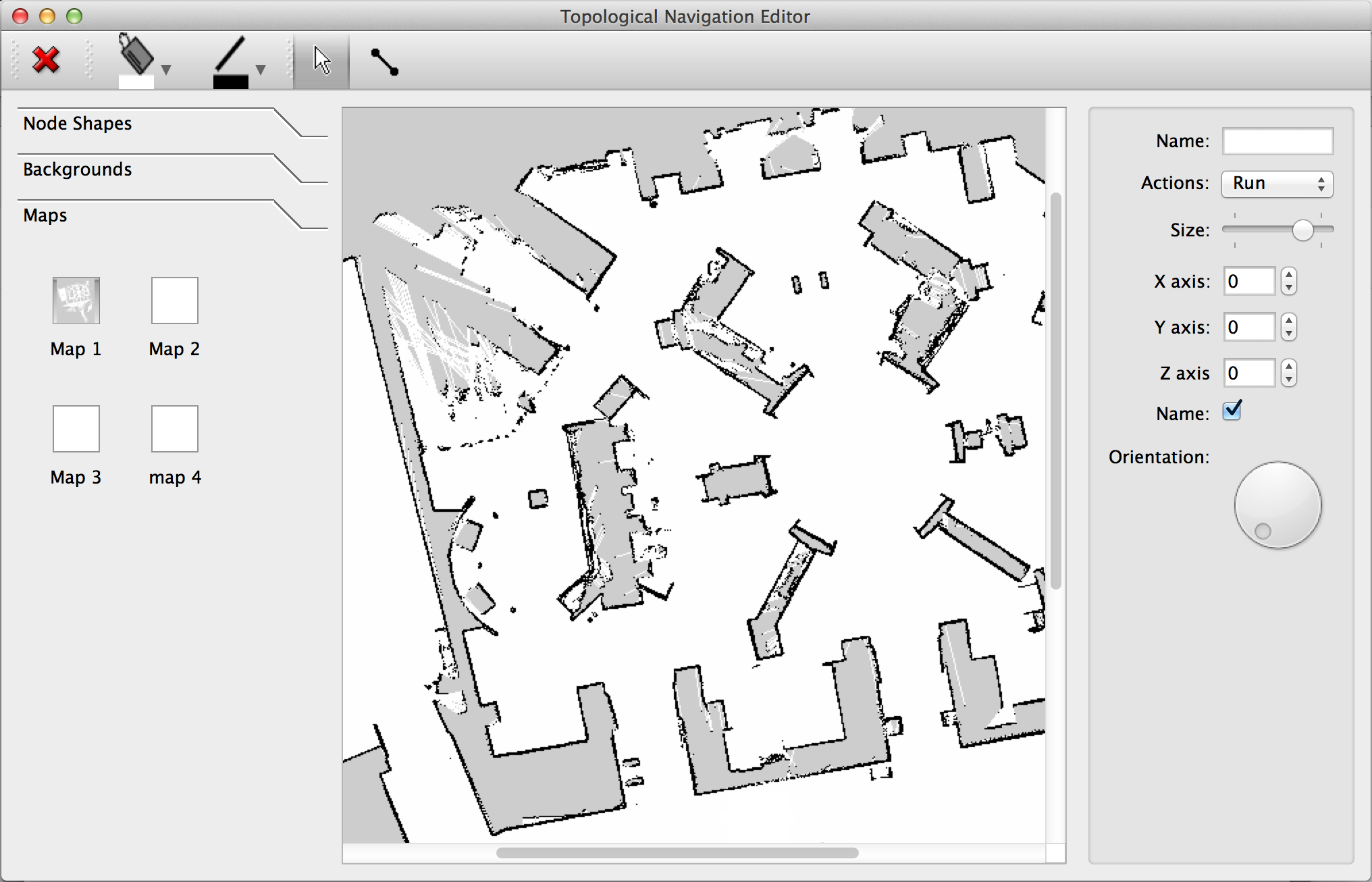
**Fifth Iteration**

Property windows



**Sixth Iteration**

Backend connection and map display



**Seventh Iteration**

Live updates/ auto-saving

Resources

**Reference:**

ISO 9241-11. (1998) *Ergonomics of Human System Interaction - Part 11: Usability: Definitions and concepts*.

Kent Beck and Cynthia Andres (2005) *Extreme Programming Explained, Embrace Change.* Pearson Education, Inc.

Jenny Preece, Yvonne Rogers, Helen Sharp (2011) *Interaction Design beyond human-computer interaction 3rd Edition*. Wiley.

SOMMERVILLE, I. 2007. *Software Engineering*. England, Pearson Education Limited.

SUMMERFIELD, M. 2007. *Rapid GUI Programming with Python and Qt*. Michigan, Edwards Brothers.

William B, Fox. (2005) *Entrepreneur, About an industry virtually guaranteed to become a Next New Thing.*

<http://www.amfir.com/AmFir/Commentary%20Pages/EditorialsArticles/I,%20Robot%20Investor/I,%20RobotEntrepPart-1-FutureGrowthTrend.html> [Accessed on 16 Jan. 16]

The Qt Company Ltd, (2015). *What is PyQt?*

<https://riverbankcomputing.com/software/pyqt/intro> [Accessed on 11 Jan. 15]

<https://books.google.co.uk/books?id=bt3oAgAAQBAJ&pg=PT228&lpg=PT228&dq=edges+in+pyqt+python&source=bl&ots=gYgs3TFZmw&sig=yZ0l8gRKPB80veQtItVLGho9gpA&hl=en&sa=X&ved=0ahUKEwj5t6796tfJAhWIVhoKHfS4AAM4ChDoAQgpMAI#v=onepage&q=edges%20in%20pyqt%20python&f=false> [Accessed on 13 Dec. 15]

Elastic nodes (custom made)

<http://ftp.ics.uci.edu/pub/centos0/ics-custom-build/BUILD/PyQt-x11-gpl-4.7.2/examples/graphicsview/elasticnodes.py> [Accessed on 12 Dec. 15]

Elastic nodes

<https://github.com/Werkov/PyQt4/blob/master/examples/graphicsview/elasticnodes.py> [Accessed on 12 Dec. 15]

Widgets

Jan Bodnar

<http://zetcode.com/gui/pyqt4/widgets/> [Accessed on 02 Dec. 15]

<http://stackoverflow.com/questions/18428095/qt4-qmenu-addaction-connect-function-with-arguments> [Accessed on 11 Jan. 16]