**FSM Generator**

**Software Detailed Design**

# Contributors

Chris Loos

Justin Butler

# Data Design

# Node design

Attempt

Anchor point

double cost

Anchor anchor

boolean isAcceptState

String text

StackPane pane

# Link design

Node start

Node end

String text

double textPadding

double parallelAngle

double perpendicularAngle

double arcAngle

Anchor startAnchor

Anchor endAnchor

Anchor arcAnchor

Node linePath

Arc arcPath

# Anchor design

double x

double y

# Settings design

double nodeRadius

Paint stateColor

Paint acceptStateColor

Paint linkColor

Paint selfLinkColor

Paint textColor

# NodePair design

String startNode

String endNode

# Node store design

HashMap<String, Node> nodeStore

# Link store design

HashMap<NodePair, Link> linkStore

# XML export design

<FSM>

<NODE name=”” accept=””>

<ANCHOR>

<X> </X>

<Y> </Y>

</ANCHOR>

</NODE>

</FSM>

# FSM export design

FSM

accept <>

a A linkName functionName

b B linkName functionName

procedures

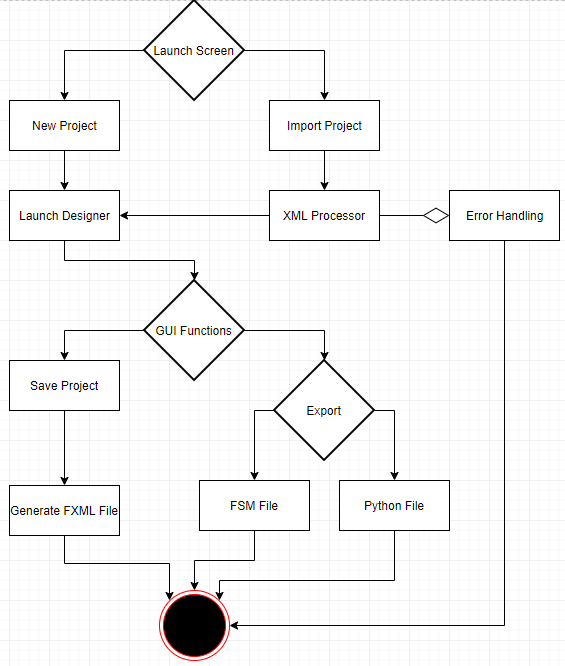
Function() {}

Function() {}

# Python export design

This file will export a transpiled version of the FSM design consisting of IF ELSE statements following Python syntax.

# Architecture design



# Interface design

# Launch screen

Contains options for creating a new project or importing an existing project.

# Menu lists

Contains options for creating new projects, importing existing projects, exporting projects, and

Closing projects.

# Designer

Contains options for creating new state nodes, new links, and modifying them.