

= XCode Project Setup after creating the actual XCode Project:

install [git](#) e.g. from <http://code.google.com/p/git-osx-installer/> # get [Doxygen](#), e.g. [the DMG](#) # get [uncrustify](#) e.g. bundled with from [UniversalIndentGUI](#)

== App ID # [Apple Developer Program Portal](#) Hint: insert one wildcard to allow multiple packages (App, Test, etc.) - so be both specific (domain, project) and liberal at the same time.

The range spanned by the App ID can be signed with **one** single Provisioning Profile, so I suggest to be specific up to a project level but allow multiple Apps to be able to try things out in small experimental Apps. == Provisioning Profile # [Apple Developer Program Portal](#) # download it and drop it into the XCode Organizer # Project → Edit Project Settings → Build → **each configuration separately** → Code Signing Identity

== .gitignore

[.gitignore](#)

```
*~
build/
.~lock*
*.modelv3
# maybe version Custom Executable Settings?
# http://cocoawithlove.com/2009/12/sample-iphone-application-with-
*.pbxuser
```

== Makefile copy alongside the *.xcodeproj and edit top 10 Lines:

[Makefile](#)

```
#!/usr/bin/make
# Make help:
http://www.gnu.org/software/make/manual/html_node/Phony-Targets.html#Phony-Targets

APP          := <Enter *.xcodeproj name without suffix>
APP_TARGET   := <Enter main target>
PROVISION    := <Enter relative path to Provisioning Profile>
DEPLOY_DIR   := <Enter (evtl. absolute) path to IPA deployment dir>
INFO_PLIST   := <Enter full Info.plist file name>
TEST_TARGET  := LogicTests

COREDATA_MODEL := $(APP).xcdatamodel

BUILD_DIR     := ./build/

VERSION       := $(shell grep -A1 CFBundleVersion $(INFO_PLIST) | tail -n 1 | egrep -oe ">([<]+)" | cut -c 2-)
DOXYGEN_DIR   := $(BUILD_DIR)/doxygen/
```

```

# non-standard tools. Set if unset in environment
DOXYGEN      ?=
$(HOME)/Applications/Doxygen.app/Contents/Resources/doxygen#
http://www.stack.nl/~dimitri/doxygen/download.html#latestsrc
DOT_PATH     ?= $(dir $(DOXYGEN))#
http://www.stack.nl/~dimitri/doxygen/download.html#latestsrc
GIT          ?= git                #
http://code.google.com/p/git-osx-installer/
OPTIPNG      ?= optipng            # http://optipng.darwinports.com/
UNCRUSTIFY   ?=
$(HOME)/Applications/UniversalIndentGUI/indenters/uncrustify    #
http://universalindent.sourceforge.net/index.php
UNCRUSTIFY_OPTS := -l OC --replace --no-backup -c uncrustify.cfg

XCODEBUILD   := xcodebuild
AGVTOOL      := agvtool
ZIP          ?= zip

.PHONY: tests pretty clean

defaultTarget: tests

SOURCES      := *.pch Classes/*.h Classes/*.m Classes/**/*.h
Classes/**/*.m Tests/*.h Tests/*.m Tests/**/*.m

#####
##
## help
##
## explain this Makefile's major targets
##
help:
    @egrep -A1 -e "^[a-zA-Z0-9]+:|^##" Makefile

#####
##
## toolversions
##
## print all used tools' versions
##
toolversions:
    @echo "-----"
    make --version
    @echo "-----"
    -$(XCODEBUILD) -version
    @echo "-----"
    -$(GIT) --version
    @echo "-----"
    -$(DOXYGEN) --version
    @echo "-----"
    -$(DOT_PATH)/dot -V

```

```

@echo "-----"
-$(UNCRUSTIFY) --version
@echo "-----"
-$(OPTIPNG) --version
@echo "-----"
-$(ZIP) --version

#####
##
## all
##
## Build anything
##
all:    pretty debug release docset

#####
##
## debug
##
## $(XCODEBUILD) the Debug / $(DEVICE_SDK) configuration
##
debug:  $(SOURCES)
        $(XCODEBUILD) -project $(APP).xcodeproj -target $(APP_TARGET) -
configuration Debug -sdk $(DEVICE_SDK)

#####
##
## release
##
## $(XCODEBUILD) the Release / $(DEVICE_SDK) configuration
##
release:  $(SOURCES)
        $(XCODEBUILD) -project $(APP).xcodeproj -target $(APP_TARGET) -
configuration Release -sdk $(DEVICE_SDK)

#####
##
## alltargets
##
## $(XCODEBUILD) the Debug / $(DEVICE_SDK) configuration
##
alltargets:  $(SOURCES)
        $(XCODEBUILD) -project $(APP).xcodeproj -alltargets -configuration
Debug -sdk $(DEVICE_SDK)

#####
##
## tests
##
## $(XCODEBUILD) & run the unit (logic) tests
##

```

```

tests: $(SOURCES)
    $(XCODEBUILD) -project $(APP).xcodeproj -target $(TEST_TARGET) -
configuration Debug -sdk $(SIMULATOR_SDK)

#####
##
## clean
##
## remove built artifacts
##
clean:
    rm -r $(BUILD_DIR)
    mkdir $(BUILD_DIR)

#####
##
## pretty
##
## indent source code (pretty print)
##
pretty:    uncrustify

uncrustify:    $(SOURCES)
    @${UNCRUSTIFY} ${UNCRUSTIFY_OPTS} `ls $? 2> /dev/null`

optipng:    *.png Images/*.png
    $(OPTIPNG) -o9 $?

#####
##
## doc
##
## build doxygen documentation. Inject version?
##
doc: $(DOXYGEN_DIR)/html/index.html

$(DOXYGEN_DIR)/html/index.html: $(BUILD_DIR)/doxygen.generated
$(SOURCES)
    $(DOXYGEN) $<
    open $@
    @echo

=====
@echo == doxygen warnings:
# @cat $(BUILD_DIR)/doxygen/warnings.log

$(BUILD_DIR)/doxygen.generated: doxygen.config $(INFO_PLIST)
    -mkdir -p $(DOXYGEN_DIR)
    cp $< $@
    sed -i '' -e "s|DOT_PATH *=.*|DOT_PATH          = $(DOT_PATH) |g" $@
    sed -i '' -e "s|PROJECT_NUMBER *=.*|PROJECT_NUMBER      ="

```

```

$(VERSION)/g" $@
    sed -i '' -e "s|OUTPUT_DIRECTORY *=.*|OUTPUT_DIRECTORY      =
$(DOXYGEN_DIR)|g" $@

#####
##
## docset
##
## build + integrate doxygen docs into XCode.
## http://developer.apple.com/tools/creatingdocsetswithdoxygen.html
##
docset: $(DOXYGEN_DIR)/html/index.html
    make -C $(dir $<) install

#####
##
## ipa
##
## build, test, tag, package into 2 ipa files ready for testing.
##
ipa:  debug release tests tag \
    $(BUILD_DIR)/ipa/Debug/$(APP_TARGET).ipa \
    $(BUILD_DIR)/ipa/Release/$(APP_TARGET).ipa

$(BUILD_DIR)/ipa/Debug/$(APP_TARGET).ipa:  $(BUILD_DIR)/Debug-
iphoneos/$(APP_TARGET).app
    -rm -r $(dir $@)
    mkdir -p $(dir $@)/Payload
    cp -rp $<.dSYM $(dir $@)
    cp -rp $< $(dir $@)/Payload/
    cd $(dir $@) ; \
    $(ZIP) -qrp9 $(APP_TARGET).ipa Payload/$(APP_TARGET).app
    rm -r $(dir $@)/Payload
    ls $@

$(BUILD_DIR)/ipa/Release/$(APP_TARGET).ipa: $(BUILD_DIR)/Release-
iphoneos/$(APP_TARGET).app
    -rm -r $(dir $@)
    mkdir -p $(dir $@)/Payload
    cp -rp $<.dSYM $(dir $@)
    cp -rp $< $(dir $@)/Payload/
    cd $(dir $@) ; \
    $(ZIP) -qrp9 $(APP_TARGET).ipa Payload/$(APP_TARGET).app
    rm -r $(dir $@)/Payload
    ls $@

#####
##
## dropbox
##
## build IPAs & doc and copy to dropbox

```

```
##
dropbox:  ipa doc
  -rm -r $(DEPLOY_DIR)/v$(VERSION) 2> /dev/null
  -rm -r $(DEPLOY_DIR)/ipa 2> /dev/null
  cp -rp $(BUILD_DIR)/ipa $(DEPLOY_DIR)
  cp -p $(PROVISION) $(DEPLOY_DIR)/ipa
  cp -rp $(DOXYGEN_DIR)/* $(DEPLOY_DIR)/ipa
  mv $(DEPLOY_DIR)/ipa $(DEPLOY_DIR)/v$(VERSION)
  ls -dl $(DEPLOY_DIR)/v$(VERSION)/*$(APP_TARGET).*

# #####
# internal use only:
# - fail if there's uncommitted changes waiting
# - fail if unable to set the Info.plist's version's tag
tag:
ifeq (2, $(strip $(shell $(GIT) status | wc -l)))
  $(AGVTOOL) next-version -all
# sed -i '' -e "s/PROJECT_NUMBER *=.*/PROJECT_NUMBER
$(VERSION)/g" doxygen.config
  $(GIT) commit -a -m "agvtool next-version: v$(VERSION)"
  $(GIT) tag -a -m "agvtool next-version" v$(VERSION)
else
  $(GIT) status      # assumed to return exit code 1 in any case.
  exit 1             # just to be 100% sure to exit in error...
endif
```

== Doxygen copy and edit the doxygen.config: # PROJECT_NAME # DOCSET_FEEDNAME #
 DOCSET_BUNDLE_ID == Apple Generic Versioning or agvtool:
<http://chanson.livejournal.com/125568.html> # Project → Edit Project Settings → Versioning System =
 „Apple Generic“ # Project → Edit Project Settings → Current Project Version = „1“ # Info.plist →
 CFBundleVersion = „1“ # Info.plist → CFBundleShortVersionString = „0.0.1alpha“ == Unit Tests
<http://cocoawithlove.com/2009/12/sample-iphone-application-with-complete.html> # Project → New
 Target... → Unit Test Bundle → „LogicTests“ # Targets → LogicTests → Get Info → General → Direct
 Dependencies → add App „main“ target # evtl. add OCMock + Project → Edit Project Settings → Build
 → Search Paths → Header Search Paths → „Classes“ (non-recursive) # Project → Edit Project Settings
 → Build → GCC_PREFIX_HEADER → MyProject_Prefix.pch # Debugging them: Project → New Custom
 Executable → Name: „LogicTestsDebug“, Path: „Developer/usr/bin/otest“ („Relative to Current SDK“) #
 Executable → Arguments → Argument

```
-SenTest Self
$(SRCROOT)/$(BUILT_PRODUCTS_DIR)/LogicTests.octest
```

Executable → Arguments → Variables

```
DYLD_ROOT_PATH = $(SDKROOT)
IPHONE_SIMULATOR_ROOT = $(SDKROOT)
DYLD_LIBRARY_PATH = $(BUILT_PRODUCTS_DIR):$(SDKROOT)
DYLD_FRAMEWORK_PATH = $(BUILT_PRODUCTS_DIR):$(SDKROOT)
CFFIXED_USER_HOME = $(HOME)/Library/Application Support/iPhone
Simulator/User
```

```
DYLD_NEW_LOCAL_SHARED_REGIONS = YES  
DYLD_NO_FIX_PREBINDING = YES  
DYLD_FORCE_FLAT_NAMESPACE = YES
```

Caution: the settings 6. + 7. go into the per-user username.pbxuser == Integration Tests todo, see <http://cocoawithlove.com/2009/12/sample-iphone-application-with-complete.html>

[XCode](#), [iPhone](#), [Makefile](#), [Doxygen](#), [agvtool](#), [Unit](#), [Test](#), [OCMock](#), [SenTest](#), [git](#)

From:

<http://wiki.mro.name/> - 

Permanent link:

http://wiki.mro.name/iphone/xcode_project_setup?rev=1272962156

Last update: **2010/05/04 10:35**

