**Guide to Preparing the**

**Maintenance Manual**

CSc 191 Senior Project

Department of Computer Science - College of Engineering and Computer Science

California State University, Sacramento

Version 2 Nov. 2021

***FRONT PAGE***

*All CSc 190 and CSc 191 documents must have the same Front Page design and format … but also include the Project name, the Team name and the list of members*

**Maintenance Manual**

**Project name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Team name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Team Members**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Date**: xx/xx/xxxx

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
| Insert table of contents – Search for how to format word docs to auto-create and update this for you. Don’t manually do it. |  |
|  |  |

1. **Introduction**

Brief description of the purpose and use of the software application. Typically a 2-5 sentences.

1. **Implementation tools and technologies**

Identify the knowledge of what tools and technologies would be needed to maintain the software product.

For example:

Programming languages

Technologies, libraries and frameworks

Web Services: Apache CXF, Axis, SOAP, WSDL, JAXB, JAX-WS

Web technologies: XHTML, HTML5, XML, XSL, XSL-FO, XSLT, CSS, XPath, XQuery, SAX,

IDE: Eclipse, Idea, MS Visual Studio, Aptana Studio, XCode, NetBeans, Komodo

Networking protocols and data security

Databases/Data storages

Application and web servers

AWS, Google, Azure

Payment services

SSO (Google, FB, etc…)

Powershell, Putty, WinSCP, other?

Account Usernames and Emails (don’t include passwords – find some other way to provide to the client)

1. **Runbook**
2. List what to do if:
   1. Server goes down
      1. How do they restart/recover?
   2. Database becomes corrupt
      1. How to restore from backup
   3. Third party services go down
      1. There may be nothing they can do until the service is back up BUT:
         1. How do they identify that this happened?
         2. What errors will they see?
            1. Ensure you code has proper error handling to convey what has gone wrong.
3. How to configure a computer for development work
   1. What needs to be installed?
4. How to make a code change
   1. Ex (show in detail):
      1. Create a branch
      2. Make changes
      3. Merge back in
      4. Undo changes
      5. Switch branches
5. How to get to the log files and examine them for errors
6. How to deploy releases
   1. They may never do it themselves, but they will need to give this to someone else to fix and deploy
7. How to run automated tests
8. How to do “sanity” check to ensure code changes didn’t break anything
9. **User interaction**

For each “screen”:

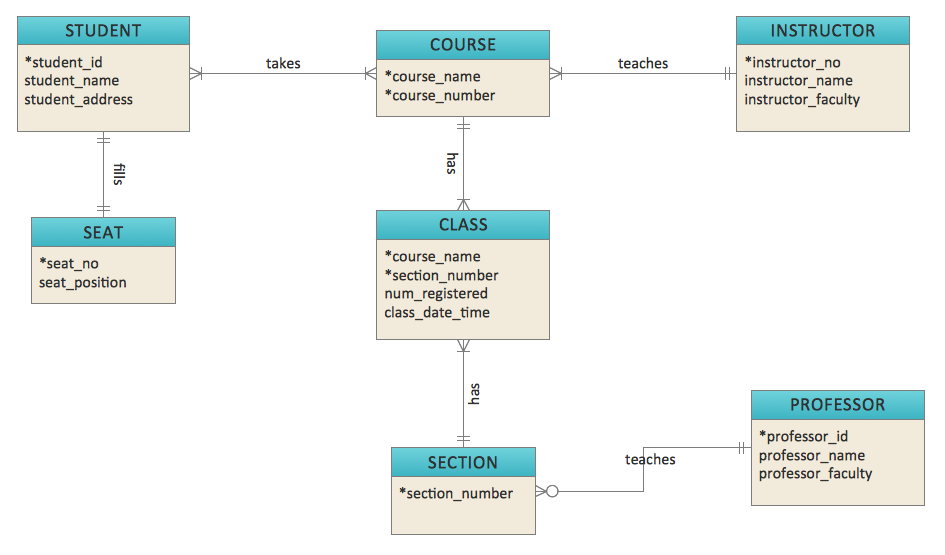
Include a “screenshot”

What types of users see the screen (admin, customer, etc)?

List code files for major components – This aids in quick debugging/enhancements in the future.

1. **Database**

List each table and its elements



**Appendices**

Anything else you wish to include

|  |  |
| --- | --- |
|  |  |