|  |
| --- |
| TeamEffort |
| Soccer Team Management System |
| Milestone 1 |
|  |
| **Simon Fanner – saf725, Patrick Weckworth – paw818, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |
| **2/2/2012** |

|  |
| --- |
| [Type the abstract of the document here. The abstract is typically a short summary of the contents of the document. Type the abstract of the document here. The abstract is typically a short summary of the contents of the document.] |

# 4. Risk

## 4.1 List of Risks

Software Requirement Risks

* Change of requirements
* Poor definition of requirements
* Impossible requirements

Software Risks

* Project & Milestone completion dates being unrealistic
* Hardware (server issues)
* Lack of Testing
* Personal differences in design/coding techniques
* Lack of knowledge could make some features unobtainable
* Human Errors

Software Scheduling Risks

* Over-estimate time requirements
* Under-estimate time requirements
* Not managing time affectively
* Requirements changing and not being able to adequately allocate time
* Lack of skill could require additional learning to implement goals
* Tool failure, like SVN, or difficulties with NetBeans or Java Server

Software Quality Risks

* Improper or lack of design documentation
* Unrealistic scheduling leading to lack of testing and deploying bug filled application
* Lack of knowledge leading to unforeseen bugs, errors, or unexpected results
* Application’s user interface not easy to use

Team Risks

* Lack of communication
* Scheduling conflicts
* Lack of responsibility (ownership)

Software Business Risks

* No one wants the application
* Budget failure (time or financial)
* Distribution failure

## 4.2 Risk Report

In light of our possible risks, the team will take a number of actions to ensure maximum risk avoidance. The requirements will be outlined as complete as possible in the first Milestone, however, Team Effort will remain flexible in case new requirements, or requirement definitions are changed. Any requirements that are deemed to be too expensive or impossible to meet will be abandoned immediately and no more time will be allocated to those requirements. Team Effort’s leader will ensure the team stays on task and manages time affectively. Team Effort will always conservatively budget time to ensure time-cost over-runs are minimal. The team will also ensure sufficient time is allocated for testing. Through code reviews and TXL’s ‘pretty printing’ the software’s code, although written by multiple programmers, will all be uniform and conform to one single style. Team Effort will take precautions to ensure all of their data and code are backed up, database through MySQL’s dump feature, and code and documents through SVN and individual checkout’s. Since the server can easily be run locally on Glass Fish or Tomcat, there is little worry about server failure as long as the project’s source code and database information is backed up. The team’s knowledge is limited, but with nine team members with diverse backgrounds the knowledge is extensive and broad. To fix or prevent future future risks from turning into failures, Team Effort will conduct formal and informal reviews of; code, documentation, time management, team communication and coherence. With reviews the team will be able to modify its behaviour and adapt before failures occur. Through all of the actions outlined here, Team Effort will minimize risk.