**CS673 Software Engineering** 

**TV Bums - BUMTV**

**Software Test Document**

| Team Member | Role(s) | Signature | Date |
| --- | --- | --- | --- |
| Zhe Huang | QA | *Zhe Huang* | 09/25 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Revision history**

| **Version** | **Author** | **Date** | **Change** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

[Testing Summary](#_heading=h.gjdgxs)

[Manuel Tests Reports](#_heading=h.30j0zll)

[Automated Testing Reports](#_heading=h.1fob9te)

[Testing Metrics](#_heading=h.3znysh7)

[References](#_heading=h.2et92p0)

[Glossary](#_heading=h.tyjcwt)

# Testing Summary

In this section, you will summarize what was tested, who is involved in testing, testing techniques used, and testing result. You may have the following tests

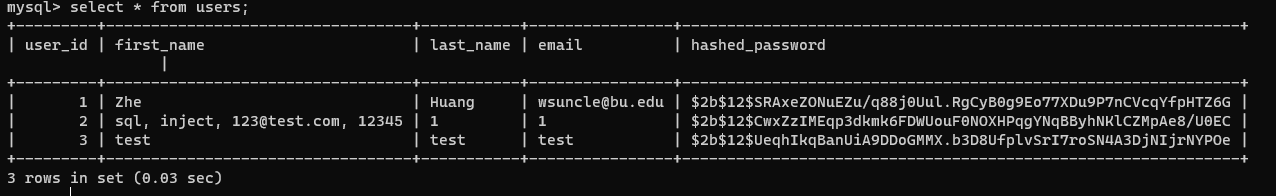
* + Unit Testing
  + Integration testing
  + System Testing
  + Acceptance Testing
  + Regression Testing

# Manual Testing Report

In this section, you will give a detailed description of each manual test case performed and the result. If this is a previous You shall list what are existing tests developed in the previous semester and what are new tests developed currently.

Here is a sample template that can be used for each test case. For system tests or acceptance tests, you may also include some screenshots.

* Test case ID： iter1 01
* New or old: new
* Test items: Login sql injection
* Test priority: High
* Dependencies (to other test case/requirement if any):
* Preconditions: (if any)
* input data: firstname : sql, inject, 123@test.com,12345) #
* Test steps:1
* Postconditions:
* Expected output: block the sql injection
* Actual output: blocked



* Pass or Fail:pass
* Bug id/link: none
* Additional notes: mysql blocked the sql injection

# Automated Testing Report

Describe briefly the automated testing you have done, including where the test code resides in your code repository, what test frameworks are used, and the screen shots or generated testing report.

# Testing Metrics

In this section, you shall report any metrics used for the evaluation, e.g. # of test cases, test coverage, defects rate, etc.

## 

| Metric Name | Description |
| --- | --- |
| LOC | We have no requirement of LOC |
| # of Files | No requirement of number of files |
| # of Classes | Quantify the total number of classes used |
| # of methods | Quantify the total number of methods/functions used. |
| cyclomatic complexity | Measure the complexity of the codebase, identifying areas that may need refactoring. |
| Defect rate (defects / KLOC) | The Defects per class |
| # of defects | Track the number of defects and how it resolved |
| # test cases | The test cases number per function/ class |
| Test case pass rate | The number that code runs well in that test case |
| Test coverage | The rate that the Test cases cover the Stories |
| Person hours used | No requirement on this |
| # of user stories completed | The number of the stories that the project can meet |

# References

# Glossary