

# Specialism Project (SDET) – Choonz

Week 12 Final project - Client Web  
Application Testing & Deployment

## Team Ice

Tay Dzonu  
Owen Jackson  
Mohammad Rashid  
Oforitsesan Towesho

06/11/2020

# Presentation

1. Project description
2. Risk Assessment
3. Sprint
4. Demo
5. Testing
6. Improvement
7. Our Journey



# Project description

- To improve upon, and provide a full test suite for, the full-stack music library web application, **Choonz**, that we were provided with from Angelica (client).
- With utilisation of supporting **tools**, **methodologies** and **technologies** that we have developed during training.

What was our focus?

- Agile working, collaboration within the team was priority and ultimately very successful.
- Rigorous **testing**.
- MVP. A **product/sprint review** with Angelica at the end of sprint 1 provided us with useful feedback on what to work towards:
  - User experience
  - Functionality



# Risk Assessment

Risk description	Cause	Risk event	Likelihood	Impact	Risk rating	Action
Lack of Time	Poor PM and TM	MVP and project spec are not met	2	4	8	Time management, planning, organisation, using Jira board correctly. Forecasting accurately
Faulty equipment	Laptop breaking/crashing	Saved work could be lost	1	4	4	Git feature branch model used regularly. Back up laptop on standby
Lack of communication	Stand ups and teams not used correctly	Delays or incompletion of the project/mark scheme not met, presentation issues	3	5	15	Prepare for any possible delays and factor in time. Asking questions early on where necessary. Discussing decisions with team regularly.
Cannot use technologies properly	Lack of understanding of technologies and working with them	Incomplete or delayed project. MVP not met. Delay in sprint finish. Application contains bugs	3	4	12	Consult team, trainers, online and QA community modules. Delegate tasks to team members who are strongest with each technology. Ensure of software is up to date.
Poor internet connection during the presentation	Unreliable internet connection	Communication of ideas may be hindered making the marker lack full understanding of the project	1	5	5	Locate close to router or use ethernet cable. Make sure other people are not using up bandwidth or downloading at the same time.
Application not rigorously tested	Lack of understanding, time or practice	80% coverage not met, JMeter not used correctly, reports not created, selenium tests not running	4	4	16	Ask questions, collaborate within team, working on feature/branch model, factoring in time to complete, keeping application simple
Merge clashes in GitHub or loss of data using Git	First time with multiple people working on same repo	Merges clashing when pushing to repo, master branch not compiling, work being lost or overwritten.	4	4	16	Using feature/branch model correctly, using regular commits with useful commit messages, having admins check code before merging
Lack of understanding of the project	Not reading the project spec, not asking questions early, not collaborating in team	MVP not met, missing out on marks from mark scheme	2	4	8	Read through the project spec, ask trainers if not 100% sure. Ask team members to confirm

# Sprint Plan

- Jira
- 2x 5-day sprints + 'sprint zero'
- User Stories
- MosCow principles
- Delegation of tasks
- Meeting the MVP

EPICS Create epic X

VERSIONS

All issues

> FrontEnd Tasks

> BackEnd Tasks

> Testing Tasks

> Documentation

Issues without epics

As a developer I want there to be a burn-down chart of each sprint included in the documentation in order	Documentation	IF-20	↑	1
As a User when I view an Album I should be able to navigate to the Artist page directly from there	FrontEnd Tasks	IF-46	↑	-
As a developer I want there to be a UML and enterprise architecture diagram in the presentation in order to	Documentation	IF-34	↓	3
As a developer I want there to be a fully utilized and up to date Jira board with user stories, acceptance crit	Documentation	IF-14	↑	5
As a developer I want there to be regular commits with useful commit messages in order to keep track of th	Documentation	IF-37	↑	2
As a developer I want there to be a documents folder in Github including images, presentation, erd/uml de	Documentation	IF-44	↑	1
As a developer I want the presentation to be well put together, show full development of the project, discus	Documentation	IF-47	↑	10
As a Client, I want the code to be checked using SonarQube	Testing Tasks	IF-52	↑	-
As a developer I want there to evidence of a fully utilized feature/branch model on Github in order to keep	Documentation	IF-29	↑	2
As a User I should be able to search for tracks, albums or artists	FrontEnd Tasks	IF-24	↑	-
As a Client, I want the latency to be <2 s, at 10000 concurrent users	Testing Tasks	IF-7	↑	-
As a Client, I want the application to be soak-tested	Testing Tasks	IF-16	↑	-
As a User when I access my home screen I want to see my playlists	FrontEnd Tasks	IF-22	↑	-
As a Client, I want the website response time to be <10 ms	Testing Tasks	IF-6	↑	-
As a developer I want there to be a .gitignore, fatjar in the final version on Github in order to run the projec	Documentation	IF-36	↑	1
As a Client, I want the Backend to be tested via integration tests	Testing Tasks	IF-23	↑	-
Create a new ERD		IF-58	↑	

IF-4 / IF-47

As a developer I want the presentation to be well put together, show full development of the project, discuss the testing in full, tools/methods used, risk assessment, discuss scrum/sprints, hurdles/barriers, how we reached MVP, demo full documentation.

In Progress

Description

Add a description...

# Application demonstration



# Testing

## Backend Testing -

- Unit tests - tests how each class (Unit) works by itself
- Integration tests - tests to ensure that the integration of the class (Unit) with the rest of the code works
- Coverage test - Checking the extent of the code which is checked by the tests above
  - 97.0% of main [industry standard :  $\geq 80\%$ ]

## Frontend Testing-

- Selenium tests – Checking whether the application interface works by automating ‘users’
  - Tested whether all buttons work
  - Ensured that processes returned the expected results
    - e.g. Playlist delete button is clicked → Playlist is no longer found

## Non-functional Testing-

- JMeter tests - Observing the system's performance at different levels of concurrent users
- Sonarqube - Used to ensure code adheres to standards and to spot bugs

# Further improvements

## Functionality-

- Search Bar to find artists,albums etc.
- Ability to create public / private playlists
- A stronger frontend design and capabilities
- Security (use cookies) - instead of URL Search Parameters to keep track of logged in user
- User Experience
  - Incorporate a better method for forgotten passwords
  - Collaborative playlists

## Testing -

- Sonarqube - A lot of issues came up that were not clear.
  - Fix the code smells



# Our Consultant journey

How have we improved?

- Learnt how to use new technologies
- Agile methodology of working
- Communication and Collaborative working skills

Things we have come to understand are important?

How has our team worked together?

- Communication

# Thank you for listening!