**Project Plan**

By Glyn Kendall and Keith Pang

**Team Vision**

For tutors who need to mark group work accurately, the Student InspectorTM is a desktop application that shows team member contributions. Unlike git inspector, our product works on Google Documents and Google Sheets.

**Team Members**

Name: Glyn Kendall

Email: [gken0004@student.monash.edu](mailto:gken0004@student.monash.edu)

Role: Developer

Name: Joshua De Luca

Email: [jdel0002@student.monash.edu](mailto:jdel0002@student.monash.edu)

Role: Quality Assurance, Developer

Responsibilities:

* Evaluates the testing of features
* Has the final say on whether a feature is finished

Name: Keith Pang

Email: [kpan0004@student.monash.edu](mailto:kpan0004@student.monash.edu)

Role: Product Owner, Developer

Responsibilities:

* Act as the primary communicator with the client liaison
* Present client requirements to the team to aid in creation of product backlog
* Lead discussion in prioritization of user stories
* Aid in backlog refinement during product review

Name: Xiang Gao

Email: [xgao0001@student.monash.edu](mailto:xgao0001@student.monash.edu)

Role: Developer

Responsibilities:

* Ensure regular physical backups are made
* Upload the updated documentation to git periodically

Name: Michael Oren

Email: [more0001@student.monash.edu](mailto:more0001!@student.monash.edu)

Role: Scrum Master, Developer

Responsibilities:

* Lead and manage all meetings, including sprint plan meetings, biweekly scrums and retrospectives
* Handles conflicts within the team
* Resolve issues that stand in the way of progress

In addition to the aforementioned responsibilities and implementing features stated by user stories, all team members also have the following responsibilities:

* Becoming familiar with the Google Documents API for Python
* Review the code of other team members
* Writing test cases for features that were created by the team member as well as other members

**Process Model**

A Scrum-like methodology will be used for the creation of the Google Drive tracking application. Two sprints of length two weeks and one of length three weeks will be used.

Before the development process begins, user stories will be created and added to the product backlog to outline the features that will be required. Then, before the commencement of each sprint in the sprint plan meeting, tasks to be completed within the sprint will be moved from the product backlog to the sprint backlog, with the aim to complete these tasks by the end of the sprint. This meeting will also involve the allocation of tasks for the sprint.

At the end of the sprint, the client will be presented with a portion of the final product to evaluate and to then update or finetune the requirements of the project. The team will also meet at this stage for a retrospective to discuss what went well and what did not during the sprint.

While the team will be following most Scrum principles, some will not be followed. Scrum mandates that the team will meet to discuss the progression of the project in a stand-up meeting on a daily basis. Due to timetabling clashes, meetings will only occur biweekly. Furthermore, a hard deadline has been put in place for the completion of the project, meaning that there is no room to negotiate an extended deadline for additional features. The lengths of the sprints are also not of equal length, with the last being three weeks long.

**Definition of Done**

The following checklist must be consulted and adhered to before a feature can be considered complete.  The checklist ensures that the feature is completed to the highest possible quality whilst also ensuring that all team members have input into designing major features and voicing concerns about certain features. Once all points have been satisfied the feature can be pushed to the master repository.

Checklist

* Review carried out by each team member to ensure the feature meets the criteria it was designed to meet
* Unit testing carried out on new feature
* Major revisions to be discussed by team members
* Minor revisions do not need to have large scale discussions
* Quality assurance officer concludes whether the testing has been done to a high enough standard
* If the feature is of an acceptable quality, then it will be signed off to be pushed to the master repository
* To push to the master repository, two team members must accept the merge request on git

**Job Allocations**

The tasks to be completed by each team member in a given sprint will be allocated in this sprint plan meeting as a group with the scrum master Michael leading the discussion. These tasks will be distributed firstly based on who wants to take on the responsibility, then on who has the technical ability, availability, etc. As a last resort, the task will be given via a democratic vote.

During the sprint retrospective, each team member will discuss their thoughts on the tasks that they had completed so that the team will know how each person fared with the tasks given to them. This will make it easier to determine the task allocation for the next sprint.

**Progression Tracking**

Trello will be used to maintain a Kanban of the tasks that are to be done, in progress and completed for each sprint, along with one Kanban for the project as a whole. All team members can refer to this Kanban online.

Short stand-up meetings will take place biweekly for team members to share what has been completed since the last meeting and any issues that they may be having that are hindering progress. The team will then work together to resolve the issues brought up by these problems.

**Management Plan**

Management of backlogs and backups will be conducted through the teams google drive folder.  A physical copy of the documents will be saved on all teammates personal devices with older revisions of the files also being saved.  A file will also be placed in the git repository as another form of redundancy.

**Time Tracking**

Time spent on tasks and projects will be tracked by an online timesheet available on the team drive on google drive.  Time spent working on the task and will be recorded along with the specific tasks completed or edited as well as who worked on it. This will allow tracking of how long tasks took and better velocity estimation in the upcoming scrums.