Table 1: Revision History

Date	**Developer(s)**	**Change**
2016/09/29	Shady Nessim	Rough Draft
2016/09/29	John-Paul Dakran	Revision #1
Date2	Name(s)	Description of changes
...	...	...

SE 3XA3: Development Plan
DNA Says

Team 10, Team Name: DNA
Kareem Abdel Mesih (abdelk2)
John-Paul Dakran (dakranj)
Shady Nessim (nessimss)

1 Team Meeting Plan

- When - Wednesdays at 10:30 AM - 11:20 AM
- Where - ETB Cafe
- Frequency - Every week, unless not needed
- Roles:

  Kareem

  - Record sounds needed
  - Create shapes needed
  - Program the logic
  - Prepare meeting discussions

  John-Paul

  - LaTeX
  - Programming the User Interface

  Shady

  - Git
  - Testing the code and maintenance

- Rules for Agendas
  - Kareem will chair the meetings
  - Take turns to record minutes, written statement and homework

2 Team Communication Plan

- Git Repository: share project files
- Email/Facebook Messenger: share other files
- Phone/SMS: discussions, questions and tips

3 Team Member Roles

- Leader: Kareem
- Scribe: different person every meeting
- Knowledge and Experience (1-7):
  - Kareem- Documentation: 7, Git: 2, LaTeX: 3, Technology: 7
  - JP- Documentation: 5, Git: 5, LaTeX: 3, Technology: 6
  - Shady- Documentation: 5, Git: 4, LaTeX: 2, Technology: 5
- Roles:

  Kareem

  - Record sounds needed
  - Create shapes needed
  - Program the logic
  - Prepare meeting discussions

  John-Paul

  - LaTeX
  - Programming the User Interface

  Shady

  - Git
  - Testing the code and maintenance

4 Git Workflow Plan

- All issues will be posted in the issue tracker.
- Descriptive commit messages should be used when committing in git.
- Milestones will be set, given a date of completion and posted on git for all group members to be aware of upcoming milestones.
- Labels will be used to prioritize and organize issues and merge requests when needed.

5 Proof of Concept Demonstration Plan

- The implementation of this project should not be difficult, however the process of recording the sounds will be time consuming.
- Testing should also not be difficult, as strings will be used and compared to test the addition of new moves to the previous pattern.
- Pygame is required to develop this project. It is not hard to install if one knows which versions are compatible with the current version of Python (3.5).
- This code will run on any platform that can run Python.
- Testing with family and friends will play an integral role in determining the functionality of the program.

Overcoming Risks: With regards to compatibility issues, the latest pygame was released in 2009, which fortunately enough is compatible with the current Python version available. However Python releases a new update, it might not be compatible anymore. However, the option to download an older version of Python is available.

6 Technology

- Python will be used to develop this project.
- It will run in its basic IDE Version 3.5.
- Framework testing will be automated to test the different cases and outcomes of the game. Family and friends will test the overall functionality and performance of the game
- LaTeX will be used to generate required documents.

7 Coding Style

- Descriptive variable names
- Descriptive function names
- Consistency in spacing
  - Spaces before and after the operators
  - No spaces beside the brackets
  - One space to separate methods
  - Space before line comments
  - No spaces in between block comment and method
- Comments
  - Descriptive
  - Not too short of a sentence
  - Not everywhere, only when needed

8 Project Schedule

- Pointer to GanntProject file
- Milestones
- Roles

9 Project Review

- Reflection?
- What went well?
- What did not go well?
- Modifications to the development plan?
- Modifications to team meetings, roles and communication? Time management?