Project Management Plan for Flash Card Maniacs

**1. Overview:**

- We will be building an android application written using Java called Flash Card Mania. Flash Card Mania is an application that will make studying and testing knowledge fun for students, teachers, and anyone wanting to test their skills. This application will provide memory reinforcement by presenting a problem that the user has to solve. After solving, the user will have immediate feedback of the correct answer. The application will test on different subject matters, such as Chemistry, Math, Literature, etc.

- January 21: Project Kickoff

- January 28: Software Management Plan Review (SPMP/SCMP)

- February 18: Requirements Analysis Review (RAD)

- March 3: Sprint 1

- March 10: System Design Review (SDD)

- March 24: Object Design Review (ODD)

- April 7: Test Plan Review

- April 14: Sprint 2

**2. Definitions:**

**3. Project Organization:**

- Client: Dr. Duraisamy

- End-user: Students, teachers, and people who want to test their knowledge.

- Stakeholder:

- **Team members**: Jenna Frisch, Ryan Grissett, Steven Mota, James Slade, and Shauna Phaxay

- Team organization (subsystem, functional):

- **Team roles:**

Jenna – CEO/Tester

Ryan - Developer

James - Developer

Steven - Documentation

Shauna - Testing

**4. Managerial process plans:**

Project startup: MOU, SPMP, meeting client for the first time

Project steady state: Coding, meetings with client, testing, meetings with group, status updates

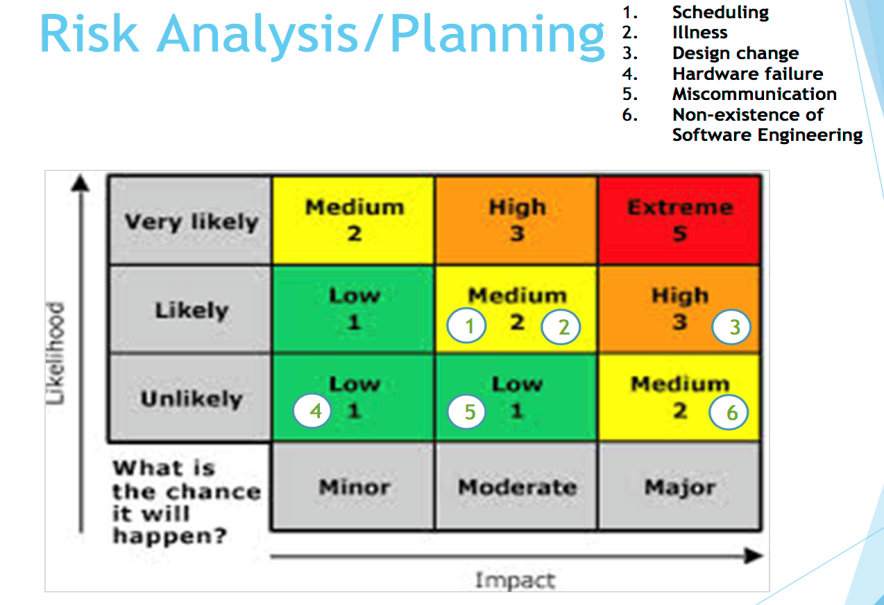
Project termination: working code, working application, meeting client's needs

Team journaling requirements: We will use Sakai to submit documents

**Risk Management Plan:**

Risk Identification:

1. People: Team members are unavailable due to scheduling.
2. People: Team members cannot complete the task due to illness.
3. Requirements: Changes to design prolong project completion.
4. Technology: Necessary documentation is lost from hardware failure.
5. Organizational: Communication failure causes workload to be increased.
6. Global: Software engineering ceases to exist and we all forget what we were doing.



**Risk Mitigation**:

1. **Scheduling** – We will minimize this risk by scheduling ahead of time to make sure everyone has plenty of time to plan. If someone cannot make it, we will assign another person to their role for the time being.
2. **Illness** – We will minimize these risks by having a backup plan of who will fill in this person’s role
3. **Design change** – We will minimize the risks that design changes could cause by making sure to ask our client’s needs ahead of time.
4. **Hardware failure** – We will make sure to save all our files onto a USB or Google Drive to minimize these risks
5. **Miscommunication** - To avoid miscommunication, we will update every person if there are changes, speak clearly, and see if anyone has any questions at the end of each meeting.
6. **Non-existence of software engineering** – Hopefully this doesn’t happen ☺

**Risk Monitoring Approach**:

Managers of the group will monitor the risks. We will draw a contingency plan to monitor these risks closely. If the likelihood of these risks increase, we will active the contingency plan.

5. **Technical Process Plans**

**Process model:** Iterative

**Development methodology:** Agile

**Development environment:**

* Object-oriented development approach
* Android Studio
* Git/GitHub
* Microsoft Project (possibly)
* USA Online (Sakai) Project Site

**Coding guidelines:**

Make good use of indentation and use meaningful names for classes, local variables, data members and methods. **Comment Comment Comment!!!!!!!!**

|  |
| --- |
|  |
|  |

6. **Supporting Process Plans:**

**Configuration management plan:**

- Identification of configuration items – Flash Card Mania

- Change control – changes will be made at the discretion of the client or developers

- Build management – This will be done by uploading code to Git/GitHub. We will all see changes made.

**Documentation plan:**

The expected documents that we are going to create and evolve are README files using GitHub. Of course the MOU and SPMP will be created. We will also have a test plan and documents for any defects found and the planned repairs on said defects. There may be designs for the user interface.

**Quality assurance plan:**

We will insure quality by following the guidelines of the project.

**Coding style:** Our coding style is going to be object-oriented.

**Security issues:** There shouldn’t be any security issues with our game.

**Reviews and audits:** We will do reviews internally by meeting Tuesday after class, specifically for that reason.