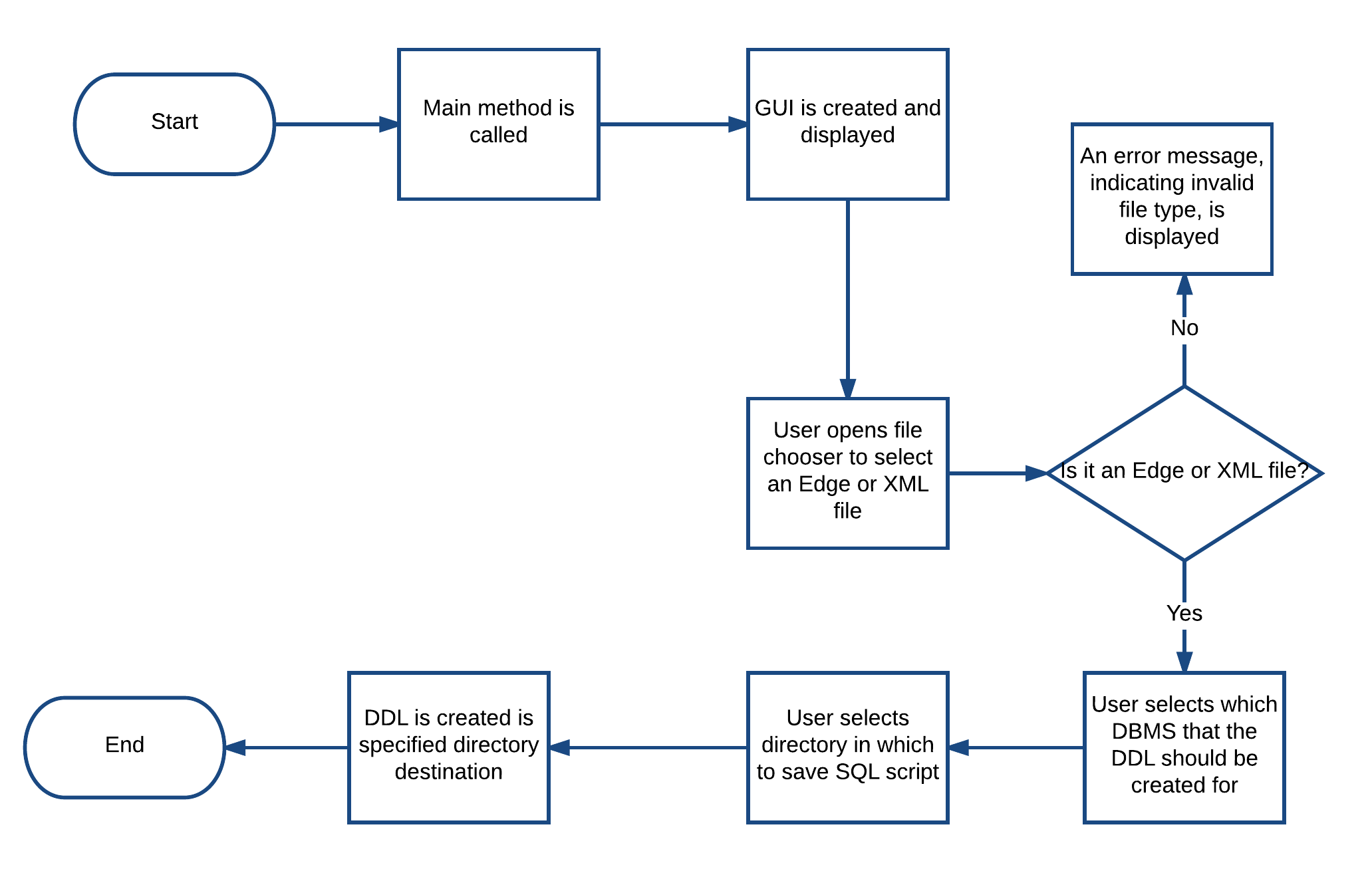
**SDLC Plan: Waterfall and Pair Programming**

Hassan Ndow, Dervent Wright, Jesus Esquivel

1. **Reasoning**
   1. Requirements are already presented upfront. They will remain unchanged throughout the development of the project.
   2. Project follows a strict, unchanged schedule. Milestones are expected to be completed and submitted by set dates.
   3. Customers are not involved in this project, eliminating the need for developers to interact or work closely with end-users.
   4. Documentation and design and are important to understanding how the system works.
   5. Each team member’s schedule does not allow for daily scrum meetings, as in an Agile environment.
   6. All developers have similar levels of programming experience.
   7. There are set periods where all developers work together with a single terminal to identify coding errors at a faster rate and to generate fewer lines of code. This expedites the development process.
   8. Testing and refactoring will be done, as needed, before moving on to the deployment stage.
2. **Milestones**
   1. **Planning**
      1. Testing Plan and Code ***✓***
      2. SDLC Plan ***✓***
      3. Work Breakdown Structure *(due 3/27/2017)*
      4. Gantt Chart *(due 3/28/2017)*
   2. **Design**
      1. System Design Documentation *(due 4/2/2017)*
   3. **Implementation**
      1. Logic Code *(due 4/18/2017)*
      2. User Interface Code *(due 4/18/2017)*
   4. **Testing / Refactoring**
      1. Unit/ Black Box Testing *(due 4/18/2017)*
      2. Refactored Code *(due 4/22/2017)*
   5. **Deployment**
      1. Help System
      2. Deployment Strategy and Packaging *(due 5/17/17)*
   6. **Other**
      1. Peer and Self Reviews
3. **Flowchart Diagram**



**Backlog Items**

* Create work breakdown structure diagram
* Create Gantt chart
* Create design documentation
* Create class(es) to read, validate, and parse XML files
* Create class(es) to generate SQL DDL output
* Create class(es) to generate user interface
* Test functionality of new classes
* Refactor classes
* Test functionality of refactored classes
* Create help system
* Create deployment plan
* Complete peer and self reviews