

**UNIVERSITY OF WATERLOO**  
Faculty of Mathematics

**#3 – Prototype Demonstration**

CS446 Sec02 Group8

Project name: AWAKE

Team name: Sleepy Devs

Team Members

Name	Quest ID	Name	Quest ID
Andrew Hua	x27hua	Eric Liu	e44liu
Yukai Wang	y3229wan	Miranda Xie	m52xie
Richella Li	j2333li	Yixing Gu	y232gu

GitHub repo link: <https://github.com/ErastoRuuuuuush/CS446>

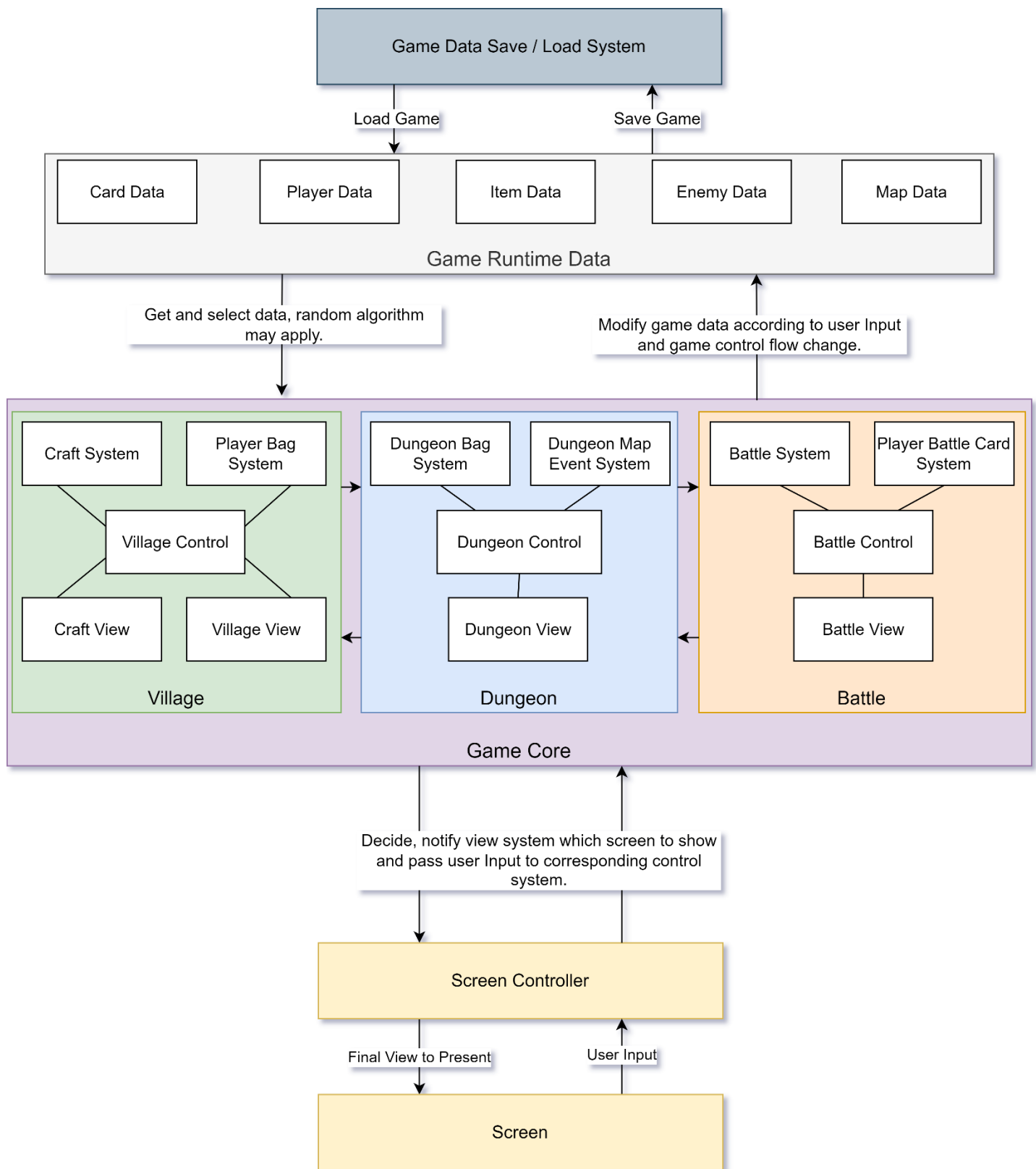
### Status Report / Demo Summary:

The implemented functionality is the battle interface of the game. The player can draw cards to attack monsters by either directly dealing damage or applying states that are recorded and causes certain effect. The users are allowed to finish a complete battle of game. The demo allows the user to select and unselect the card, to use the card by swiping on the screen (and applying the corresponding effects), to proceed through the battle rounds, to win or lose the battle, and to restart a new battle.

Currently, the graphics used are not final version and are from websites, the interaction of cards, battle controller, player, and monster battle status (HP, energy, hand, deck, etc.) are all real and well coded. The energy-restoring scheme is not yet fully determined, thus is not shown in the interfaces, but it's logic has been implemented in the backend.

For the current progress, we approximately finished 1/3 of the whole project. Our system mainly have three interface, the village interface, the dungeon interface, and the battle interface. The difficulties are that the unfamiliar use of new game framework libGDX to display and allow efficient player interaction. As next month is the last month for the project, we will need to finish the whole project. The route to finish would be:

1. Finish dungeon interface.
2. Enrich the enemy type, dungeon event type, card type.
3. Finish village interface.
4. Finish game story and its scripts, along with graphics.
5. Add tutorial for the game if time permits.



As our game is designed to play completely offline and no online interaction is needed, all data is stored and processed on the local phone device. The “Game Data Save / Load System” stores all data of the game and graphic resources when the game is not running. The “Runtime Data” and “Game Core” system will be created when the game starts running and communicate with the phone I/O for screen display and data access / storage.